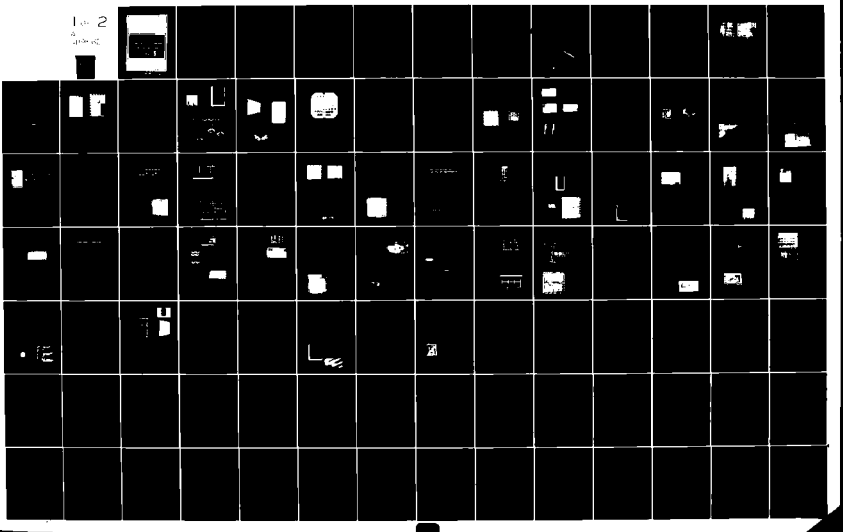
AD-A119 562 ADVISORY GROUP FOR AEROSPACE RESEARCH AND DEVELOPMENT--ETC F/G 9/2
DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTRO--ETC(U)
JUL 82 L J URBAN
UNCLASSIFIED AGARD-AR-178 NL



AGARD-AR-178

AGARD ADVISORY REPORT No. 178

Distributed Micro-Processor Applications to Guidance and Control Systems

NORTH ATLANTIC TREATY ORGANIZATION

ADVISORY GROUP FOR AEROSPACE RESEARCH AND DEVELOPMENT

(ORGANISATION DU TRAITE DE L'ATLANTIQUE NORD)

AGARD Advisory Report No.178

DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS

Edited by

Louis J.Urban
Technical Director
Deputy for Avionics Control
Aeronautical Systems Division
Wright-Patterson AFB, Ohio 45433
USA

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under the Chairmanship of Louis J.Urban.

THE MISSION OF AGARD

The mission of AGARD is to bring together the leading personalities of the NATO nations in the fields of science and technology relating to aerospace for the following purposes:

- Exchanging of scientific and technical information;
- Continuously stimulating advances in the aerospace sciences relevant to strengthening the common defence posture;
- Improving the co-operation among member nations in aerospace research and development;
- Providing scientific and technical advice and assistance to the North Atlantic Military Committee in the field of aerospace research and development;
- Rendering scientific and technical assistance, as requested, to other NATO bodies and to member nations in connection with research and development problems in the aerospace field;
- Providing assistance to member nations for the purpose of increasing their scientific and technical potential;
- Recommending effective ways for the member nations to use their research and development capabilities for the common benefit of the NATO community.

The highest authority within AGARD is the National Delegates Board consisting of officially appointed senior representatives from each member nation. The mission of AGARD is carried out through the Panels which are composed of experts appointed by the National Delegates, the Consultant and Exchange Programme and the Aerospace Applications Studies Programme. The results of AGARD work are reported to the member nations and the NATO Authorities through the AGARD series of publications of which this is one.

Participation in AGARD activities is by invitation only and is normally limited to citizens of the NATO nations.

Published July 1982

ISBN 92-835-1428-9

Printed by Technical Editing and Reproduction Ltd
5–11 Mortimer Street, London, W1N 7RH

PREFACE

WORKING GROUP 06 REPORT

Microprocessors are finding ever-increasing applications in NATO guidance and control systems. New technological developments are making even more new applications possible. Our Working Group was formed because it was becoming increasingly apparent that there is a critical need for a better understanding of the microprocessor technology developments, a better understanding of architectural configurations for NATO guidance and control systems, a need for a lexicon of terms and nomenclatures applicable to microprocessors and a need for identifying standardization options and opportunities. The four chapters of our report document our activities in each of these areas.

In the first chapter, Dr. Richard Smyth has succinctly described all areas of today's microprocessor technology. In the second chapter, Mr. Richard Bousley has expertly detailed guidance and control architectures utilizing microprocessors. In the third chapter, Dr. Thomas Cunningham has provided an excellent lexicon of microprocessor and digital systems terminology and nomenclatures. In the fourth chapter, Professor John Shepherd has made a real contribution by highlighting standardization opportunities and options applicable to microprocessors used in NATO guidance and control systems.

The first three chapters were prepared to serve as useful reference material for NATO scientific and engineering personnel. We also believe that Chapter III will serve as a stimulus for the NATO standardization community, enabling them to begin standardization of terminology and nomenclature applicable to microprocessors.

The fourth chapter of our working group report, outlining standardization options and opportunities, should be regarded as a smorgasboard or a buffet table filled with many good things. Each NATO nation may come to the table and fill their plates with those items which are appealing to their appetite and digestive system. Not everything will be, or should be, considered equally attractive to everyone. Instead, Chapter IV should be regarded as a reference containing standardization options and opportunities for future consideration. No country should feel they must accept all of the options for standardization presented.

In addition to the four aforementioned individuals who organized material given to them by other members of our working group into cohesive chapters, special thanks and recognition are due to other members of our working group who include: Mr. Richard Bogenberger (GE), Monsieur Gerard Boyer (FR) Dr. Anthony Callaway (U.K.), Dr. David W. Geyer (U.S.), Mr. Robert Hawkins (U.S.), Mr. Richard Mejzak (U.S.), Mr. Hartmut Thomas (GE), and Monsieur Jean M. Valembois (FR).

Special recognition is also due my secretary, Mrs. Agnes M. Vislosky, who handled my correspondence and assembled the final manuscript.

LOUIS J. URBAN
Technical Director
Deputy for Avionics Control
Aeronautical Systems Division
Wright-Patterson AFB, Ohio

CONTENTS

CHAPTER I
MICROPROCESSOR TECHNOLOGY

BY
DR. RICHARD K. SMYTH
MILCO INTERNATIONAL, INC.
U.S.A.

1.1.0 GENERAL

1.1.1 Background

The application of microprocessor technology to guidance and control systems has been the most important advance in this area during the last half of the 1970's. The technology advances during the last nine years when 4 bit microprocessors were first introduced by Intel (the 4004) and Rockwell (the PPS-4) have been truly astounding. Perhaps no period of time has seen such a rapid turn-over in technology. The most recent announcement of new 32 bit microprocessors challenges the fastest mainframe computers for throughput and computational capability; the new 16 bit microprocessors approach the capability of advanced minicomputers, but both at a small fraction of the cost, volume and power.

Even though the advances during the last nine years have been astounding, the promise of changes for the next decade promises to be equally amazing. There appears to be another three orders of magnitude improvement before we reach saturation limits due to inherent physical constraints, such as the speed of light limiting gate switching speeds or the tunneling of electrons from the channel to the gate.

This almost unlimited computational power enables guidance and control system mechanization approaches which were neither technically nor economically feasible before. For this reason, the NATO AGARD Guidance and Control Panel felt that it was timely to examine the impact of this mushrooming technology on guidance and control applications.

1.1.2 Objectives

The objective of Chapter I, Microprocessor Technology, is to examine the current technology state-of-the-art and identify important technology trends to permit guidance and control engineers to understand how this technology may impact their system designs in the future.

A secondary objective is to provide the guidance and control engineer an overview of the current technology to permit sound selections of appropriate technology for current designs.

1.1.3 Contributions

All members of the NATO AGARD Guidance and Control Panel Working Group 06 on Microprocessor Technology have provided helpful discussion, background material, and comments. Mr. Louis J. Urban, USAF, WG 06 Chairman, provided a summary of microcomputer types and characteristics taken from his annual update of his lecture on Avionics at the University of California at Los Angeles (Course organized by Professor Cornelius Leondes of UCLA).

The material used in the illustrations was derived from the sources listed in table 1.1.3.1 which consist of a number of magazines and periodicals reviewed over the last three years. The last source is a state-of-the-art survey of Avionics and Control sponsored by Dr. Herman A. Rediess of NASA Headquarters which was edited by the author. The contributors to that document are listed in its foreword.

TABLE 1-1.3-1
Sources of Material
(from the last two years, approximately)

- Electronics Magazine
- IEEE Spectrum
- IEEE Computer
- Aviation Week
- AIAA Aeronautics and Astronautics
- NASA Avionics and Controls State-of-the-Art CR 159050

1.2.0 MICROCIRCUIT TECHNOLOGY TRENDS

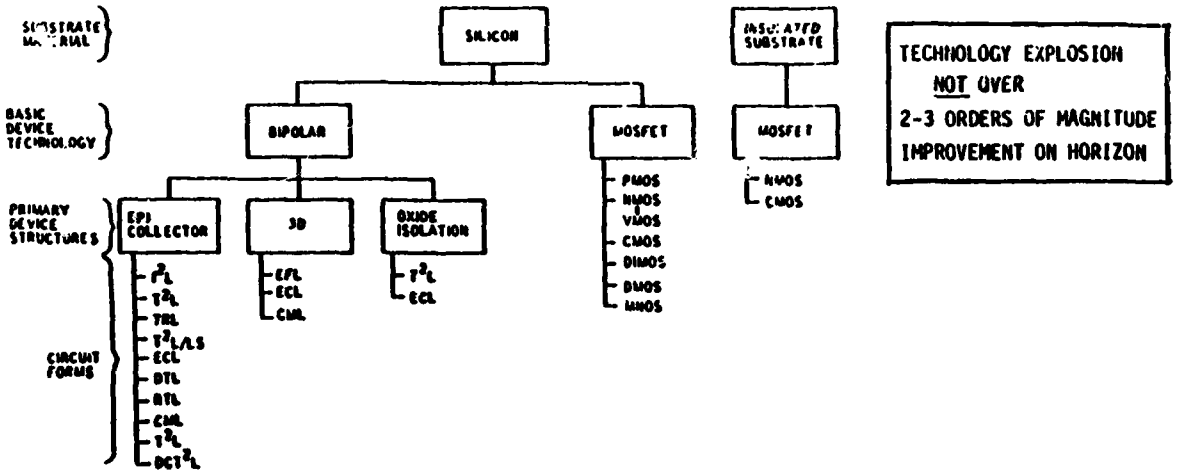
The dominant technology used in microprocessors continues to be variations of silicon technologies. Other technologies which are vying with silicon for dominance in the future include gallium arsenide (GaAs) and Josephson Junctions (JJ). The research investment in silicon technology is enormous when compared with GaAs and JJ technology. For this reason silicon should remain the dominant technology well into the 1990's. Recent advances in submicron lithography using x-rays for achieving 0.3 micron feature size promise significant improvements in the speed power products for operational microcircuits in the near future. Bell Labs has recently achieved switching speeds of 30 to 75 picosec and 5 to 50 femptoJoule power-speed product using experimental 0.3 micron silicon MOSFET circuits produced with x-ray lithography. This performance exceeds that achieved so far with GaAs operating at a 1.1 micron feature size. IBM has produced experimental JJ circuits operating faster and with a lower power product than the Bell Si MOSFETs, but the requirement for cryogenic cooling for JJ places a serious constraint on the use of this technology for airborne guidance and control systems.

1.2.1 Technology Families

The technology relationships are shown in figure I-2.1-1 for the family of silicon technologies. The family tree is broken down into substrate material, basic device technology, primary device structures, and circuit forms. The two branches in the silicon family tree are bi-polar devices and MOSFET (metal oxide semiconductor field effect transistor) devices. Currently bi-polar devices are faster, but MOSFET devices are most important for microprocessor technology. An important branch of substrate material is that of insulated substrates that provide higher switching speeds for a particular technology. For example, CMOS (complementary MOS) on sapphire substrates provides MOSFETs with very low power and high switching speeds. This technology is referred to as CMOS/SOS (silicon on sapphire). This technology receive a set back in 1981 when RCA, a pioneer in this technology decided to de-emphasize CMOS/SOS; however Rockwell (USA) and Toshiba (Japan) are still active in promoting this technology.

SEMI CONDUCTOR TECHNOLOGY TRENDS

TECHNOLOGY RELATIONSHIPS



EMERGING TECHNOLOGIES

- Ga As FET, 100 PSEC DELAY, .01 PICOJOULE/GATE (Cf WITH 1 PJ/GATE WITH I^2L)
- Si MOS FET VERY SMALL GATE SIZE e.g. 1MICRON 80PSEC DELAY, 0.1 PJ/GATE ROCKWELL REPORTS 61 STAGE RING OSCILLATOR
- JOSEPHSON JUNCTION .0001 PJ/GATE
- PROJECTING CMOS/SOS VLSI, 200MIL CHIP WITH 50-100K GATES, 300 MHz

Figure 1 2.1 1

The dominant bi-polar technology is T^2L Schottky devices. The fastest bi-polar technology is ECL (emitter coupled logic) which is pushed by Motorola and Fairchild. The Cray and Amdahl supercomputers used ECL devices with sub-nanosec switching times. The biggest drawback of ECL is the large power dissipation per gate. I^2L devices were highly promoted by Texas Instruments during the late 1970's, but this technology was not successful on the market. TI has not been pushing I^2L recently.

The dominant silicon technology by a large margin is NMOS MOSFETs which form the basis of most popular microprocessors. Variants of NMOS are VMOS by Motorola and HMOS by Intel that have geometries which provide higher circuit densities and faster gate switching than plain NMOS. The future of NMOS continues to look bright as X-ray and E-beam lithography permit submicron feature size with the attendant increases in circuit density and speed-power product.

The original 4 bit microprocessors used PMOS which is slower than NMOS. Many calculator chips still use PMOS technology because of its low cost. Its use is expected to wane in the future.

Gallium Arsenide is an emerging technology holding great promise for the future. GaAs possesses electron mobilty around 5 times greater than that of silicon, which provides an inherent advantage in speed-power product for the same feature size. Another advantage of GaAs is its high Curie point which may permit GaAs circuits to operate at 400°C. GaAs MOSFETs have operated with a gate delay of 100 picosec to yield a speed power product of .01 picoJoule (10 femptoJoules) per gate.

Permeable Base Transistor (PBT) technology is a promising new technology illustrated in figure 1.2.1.2. GaAs substrate has been utilized to achieve 30 gHz operation for PBT using 0.16 micron X-ray lithography. This technology was conceived by the MIT Lincoln Labs and is applicable to silicon as well as GaAs. Projections foresee GaAs PBT devices achieving switching frequencies of 500 gHz, corresponding to a gate switching time of 2 picosecs.

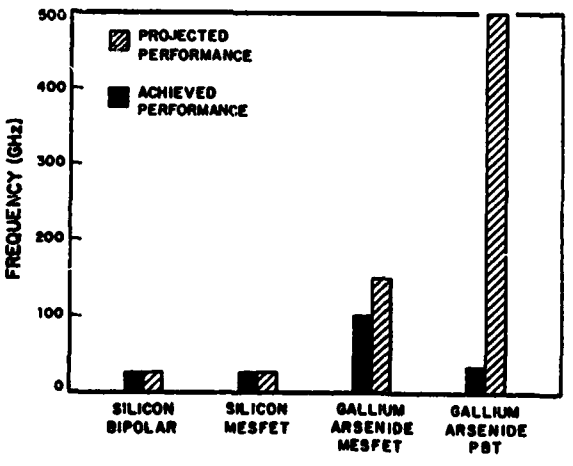
Figure 1.2.1.2

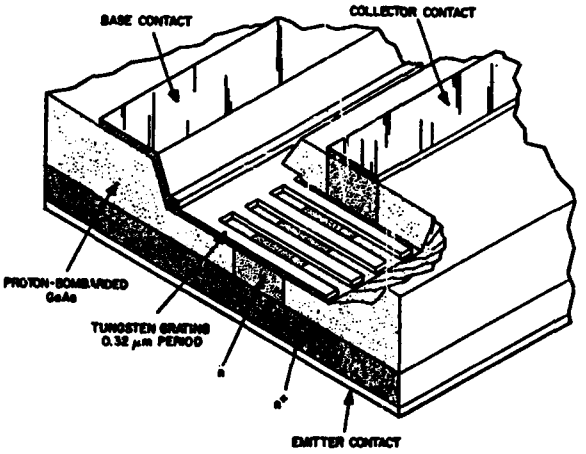
PROMISING NEW TECHNOLOGY

Permeable Base Transistors PBT

- X-Ray Lithography - 0.16 micron
- GaAS Substrate
- 30 GHz Achieved/Projected 500GHz
- Electrons Flow From Emitter to Collector Through Tungsten Fingers

  Acts Somewhat Like Triode
- Large Power Amplification in Small Area





- PBT OSC FREQ compared with Si and GaAS MESFETS and Si Bi-Polar
- Lincoln Labs working on Si PBT
- TI Developing PBT under ONR Contract

  Conceived By Lincoln Labs

1.2.2 Speed-Power Products

The use of speed-power product as a figure of merit is widely used in comparing one technology against another. Figure 1.2.2.1 illustrates this comparison figure for a number of commonly used and emerging technologies. The speed-power product consists of the power dissipation per gate times the gate switching time or propagation delay. The smaller the speed-power product, the better the technology. It should be noted that the speed-power product for a given technology is dependent upon the feature size (i.e., lithography size) selected. The diagonal lines on figure 1.2.2.1 are for constant speed-power product expressed in femptoJoules (10^{-15} Watt-sec) or picoJoules (10^{-12} Watt-sec).

The Josephson Junction technology devices have the lowest speed-power product, but JJ devices require cryogenic cooling down to about 2-3 degrees Kelvin.

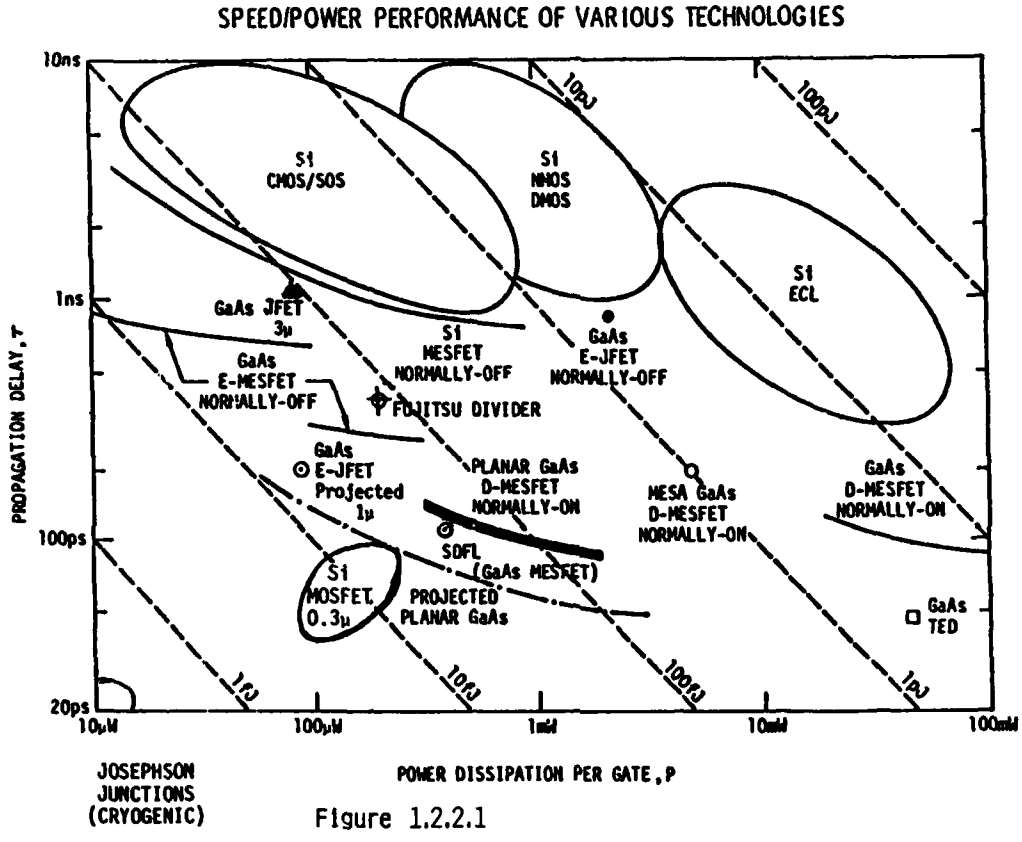


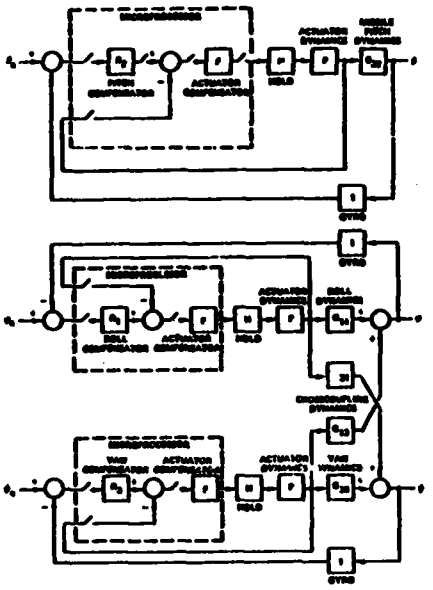
Figure 1.2.2.1

Bell Labs has recently reported a 0.3 micron Si MOSFET using X-ray lithography which achieved a speed-power product of 5 fJ. This performance exceeds that achieved by GaAs so far.

The speed of microprocessors has reached the point that the incorporation of microprocessors in digital autopilots is now common, as shown in figure 1.2.2.2.

MICROPROCESSOR IMPLEMENTED DIGITAL AUTOPILOTS COMMON PLACE

MISSILE DIGITAL AUTOPILOT



ACTUATOR AND GYRO LOOPS CLOSED IN MICROCOMPUTER

W-PLANE COMPENSATOR

$$w = \frac{z-1}{z+1} \qquad z = e^{sT}$$

Actuator	$F = \frac{(w+1)}{(w/10+1)^2}$
Pitch and yaw	$1.6J_i$ $R_2 = R_3 = \frac{(w/0.06+1)}{(w+1)}$
Roll	$0.32J_i$ $R_1 = \frac{(w/0.15+1)}{(w/2+1)(w/5+1)}$

$^aJ_i = \max(3-5t, 1)$, where t is time since launch.

12.5 MS SAMPLING RATE

16 BIT WORD LENGTH
12 BIT ADC
256 RAM/768 ROM
HIGH LEVEL LANGUAGE PROGRAMMING
(FORTRAN SUBSET)

Figure 1.2.2.2

REF. W. ALBANES, JOURNAL OF GUIDANCE AND CONTROL

1.2.3 Silicon Technology

The Soviet Union is also active in microprocessor technology. A Soviet adaptation of the Intel 8080 is shown in figure 1.2.3.1 which was a 1977 copy of a 1974 version of the 8080. The latest Intel version is somewhat smaller.

1.2.3.1 NMOS

The most widely used microprocessor technology is N-channel MOS (NMOS) or one of its variants such as HMOS, VMOS, or DMOS. Typical NMOS has a gate power dissipation of 1 mW/gate and a switching speed of 5 nanoseconds for a power speed product of 5pJ. The feature size of most current VLSI NMOS microprocessors is in the 3-4 micron range. Si MOSFETs can achieve considerably lower speed-power products by reducing the feature size into the sub-micron realm. A 0.3 to 0.4 micron NMOS silicon MOSFET produced by Bell Labs is described in figure 1.2.3.1.1.

1.2.3.2 CMOS

Complementary-MOS (CMOS) is a low power silicon technology because the quiescent power is very low when the gates are not switching. A strong trend is the conversion of NMOS microprocessors and other devices into CMOS devices to reduce system power. This trend to CMOS devices is summarized in figure 1.2.3.2.1. Typical reduction in power of a CMOS device compared with an NMOS device is greater by a factor of 10 or more.

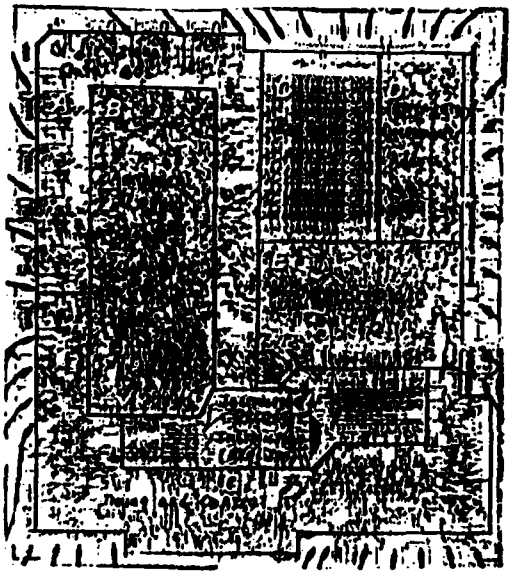
1.2.3.3 CMOS/SOS

The use of insulated substrates reduces power consumption, and increases speed while increasing tolerance to nuclear radiation. The most common insulated substrate is sapphire on which silicon is grown. Silicon on sapphire (SOS) is used with CMOS technology for a composite technology referred to as CMOS/SOS, or CMOS on sapphire substrate. In the past, RCA was a strong proponent of CMOS/SOS technology, but recently RCA set aside CMOS/SOS in favor of CMOS technology. Toshiba has recently announced a combination NMOS and CMOS/SOS 16 bit microprocessor with very low power dissipation. Most CMOS/SOS devices to date have had feature size ranging from 3-4 microns. This technology yields 100 microWatts per gate, and a gate delay of 5 nanosecs, yielding a speed-power product of about 500 femptoJoules.

Figure 1.2.3.1 TECHNOLOGY THRUST

FREE WORLD HAS NO MONOPOLY ON MICROPROCESSOR TECHNOLOGY

INTEL 8080A MICROPROCESSOR



SOVIET K580IK80 MICROPROCESSOR

(1977 Adaptation of an Intel 8080A from 1974)

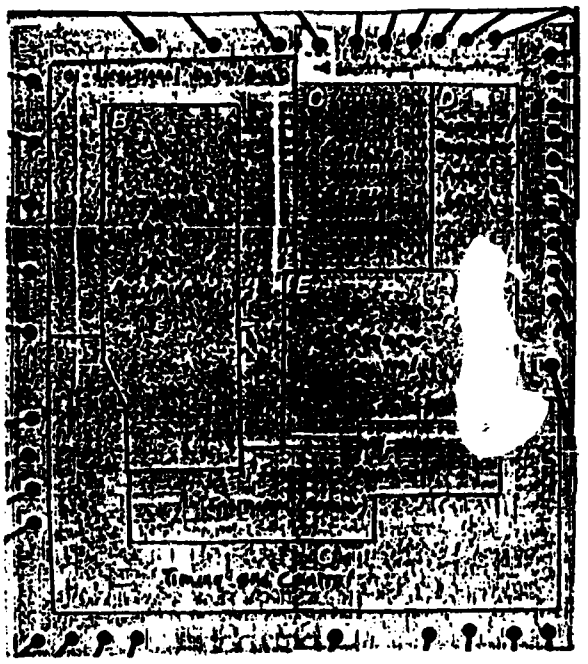


Figure 1.2.3.1.1 SUB-MICRON SILICON MOSFETs

Bell claims X-ray litho MOS 'will blow GaAs out of water'

MR. LEPSELETER
BELL LABS DIRECTOR
OF VLSI DEVICES

- BELL LABS USES X-RAY LITHOGRAPHY TO PRODUCE SILICON MOSFETS WITH 0.3 TO 0.4 MICRON CHANNEL LENGTH

  ⬇

  SWITCHING SPEEDS 40 TO 75 PICOSEC

  SPEED POWER PRODUCT 5 TO 50 FEMPTO JOULES

  10^6 MOS TRANSISTORS PER CHIP

- SPEED FASTER THAN ECL AT MUCH LESS POWER

- GaAs LOGIC SWITCHING IN 50-100 PICOSEC AT 1.1 MICRON CHANNEL LENGTH

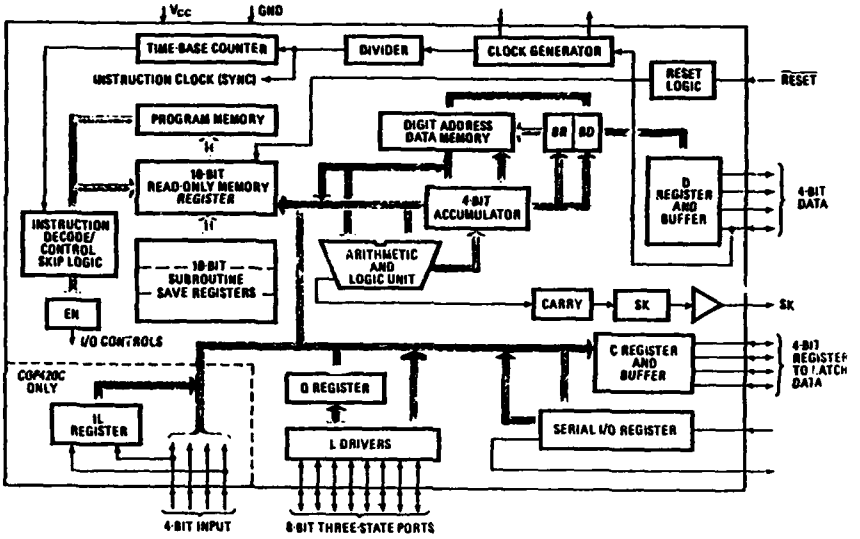
CONSENSUS OF BOTH SILICON AND GaAs PROPONENTS:

SILICON ADVANCES WILL KEEP SI AHEAD OF GAAS DURING NEXT SEVERAL YEARS

C-MOS DEVICES COMING

Figure 1.2.3.2.1

- HARRIS SELF ALIGNED JUNCTION-ISOLATED (SAJI) PROCESS
  - -CMOS VERSION OF 16 BIT 8086 MICROPROCESSOR (80C86)
  - -16K CMOS RAM
  - -CMOS VERSION OF LSI-11
  - -CMOS VERSION OF 8 BIT 8748 MICROPROCESSOR (87C48)
  - -SPEED COMPARABLE TO HMOS (INTEL's VERSION OF N-MOS)
- RCA SETS CMOS/SOS ASIDE FOR CMOS
  - -1804 MICROPROCESSOR, SINGLE CHIP VERSION OF 1802
- CMOS MICROCONTROLLER NATIONAL COP420C/320C
  - -4mw ACTIVE/50μw STDBY
  - -1 KBYTES ROM
  - -256 BITS RAM
  - -32 KHz CLOCK
  - -2.4v POWER SUPPLY
  - -$5 in QTY of 100,000
- 12 BIT AD CONVERTER
  - -BECKMAN 7581C
  - -$24.50



1.2.3.4 ECL

The emitter coupled logic (ECL) technology is the fastest silicon technology commercially available, but it is very power consuming. The gate delay is approximately 1 nanosec with power per gate of 10 mW yielding a speed-power product of 10 picoJoules or about 20 times higher than CMOS/SOS. Currently Motorola has a ECL logic family plus a 4 bit slice microprocessor family. Fairchild has an 8 bit slice microprocessor family. The Cray-1 and Amdahl-470 supercomputers use ECL technology to achieve their high speed.

1.2.4 Gallium Arsenide Technology

Gallium Arsenide (GaAs) is one of the III-V compounds which acts like a semiconductor (e.g. silicon), being a IV element. The GaAs digital devices promise significant improvements over silicon devices, because the electron mobility of GaAs is about 5 times greater than that of silicon, providing a lower speed-power product for the same lithography size as silicon.

The most serious impediment to the introduction of GaAs digital circuits is the relative low R & D investment in GaAs vis à vis Silicon technology. Also, the higher defect rate in GaAs substrate decreases the yield and causes difficulty in the fabrication of VLSI GaAs. The promising GaAs technologies are summarized in figure 1.2.4.1. The Schottky Diode FET Logic (SDFL) developed by the Rockwell Science Center has the best potential for VLSI of the technologies shown on the figure. However, the PBT GaAs technology summarized in figure 1.2.1.2 may be even better than SDFL devices. The operation of GaAs circuits above 400 degrees C is a possibility that makes these devices attractive for military and other applications requiring temperature extremes.

The MIT Lincoln Labs is working on a new method to grow GaAs substrates called 'cleft-grown GaAs thin film', shown in figure 1.2.4.2. This technique promises to reduce the surface defects for GaAs substrates and reduce the cost over melt-grown GaAs wafers.

1.2.5 Josephson Junction Technology

Josephson Junction, or JJ Technology, uses lead alloys immersed in liquid helium to achieve a few degree Kelvin junction temperature. A JJ memory chip fabricated by IBM is shown in figure 1.2.5.1 and is compared with a GaAs ring-counter fabricated by Rockwell Science Center. The JJ technology has a delay-power product of 0.5 femptoJoule with 10 to 20 pico second gate delays and 10 microWatt gate power dissipation. The JJ device has 10 times better delay-power product than the 0.3 micron Si MOSFET of Bell Laboratories shown in Figure 2.3.1.1.

FIGURE 1.2.4.1

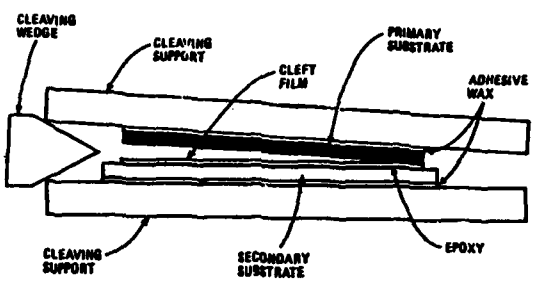
DIGITAL GaAs CIRCUIT (DIGAC) TECHNOLOGY DEVELOPMENTS PROMISE TO REVOLUTIONIZE DIGITAL AVIONICS AND CONTROLS DURING THE NEXT DECADE

- FUNCTIONS NOT POSSIBLE TO IMPLEMENT PREVIOUSLY WILL BE FEASIBLE
- SMALL SIZE & LOW POWER TREND WILL CONTINUE TO 3 ORDERS OF MAGNITUDE IMPROVEMENT
- SUPER HIGH SPEED PROCESSING
- CIRCUIT TEMPERATURE GREATER THAN 400^{o}C POSSIBLE

PROMISING GaAs TECHNOLOGIES

- SCHOTTKY DIODE FET LOGIC (SDFL)
- ENHANCED JUNCTION FIELD EFFECT TRANSITION (E JFET)
- DEPLETION METAL SEMICONDUCTOR FIELD EFFECT TRANSITION (MESFET)
- TRANSFER ELECTRON DEVICES (TED)

Figure 1.2.4.2 CLEFT GROWN GaAS THIN FILM



- Promises Low Cost, Less Surface Defects
- Cleft Film Wafer 3-4 microns thick

  --- Melt grown wafers 300-400 microns thick

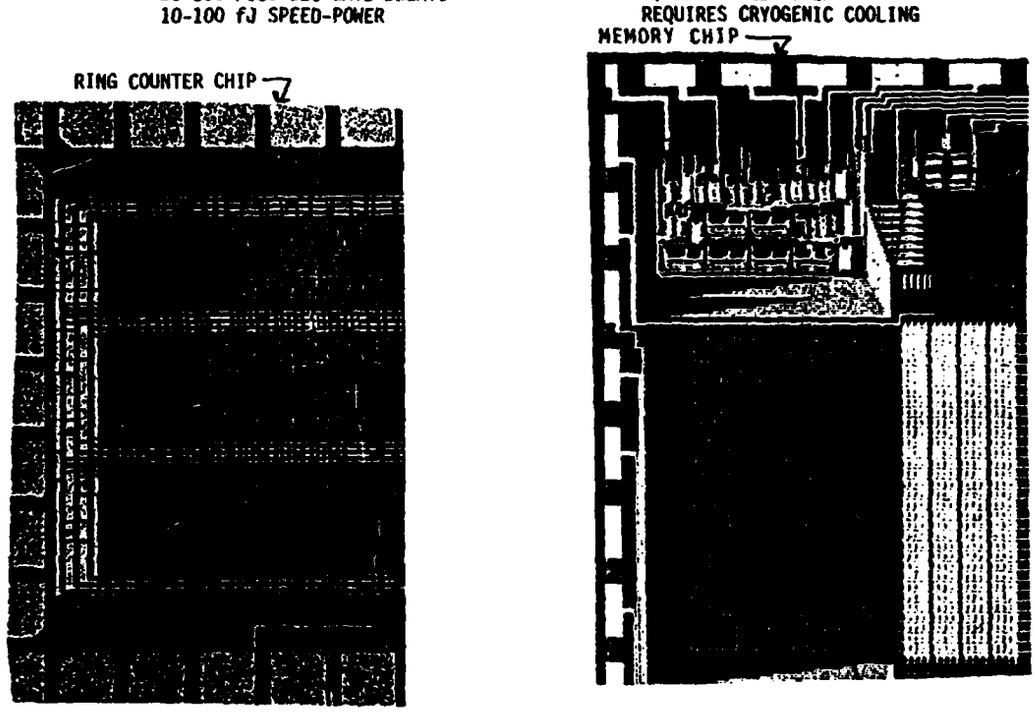
- MIT Lincoln Labs

Figure 1.2.5.1

ALTERNATE CIRCUIT TECHNOLOGIES TO SILICON

GALLIUM ARSENIDE (III-V COMPOUND.,
4-5 TIMES ELECTRON MOBILITY OF SI
80-100 PICO SEC GATE DELAYS
10-100 fJ SPEED-POWER

JOSEPHSON JUNCTION (LEAD ALLOYS)
10-20 PICO SEC GATE DELAYS (MIN 1 PICO SEC)
0.5 fJ SPEED-POWER
REQUIRES CRYOGENIC COOLING



1.2.6 Component Density Factors

The lithography used to fabricate microprocessor chips limits the line widths and other feature sizes such as the gate channel widths. The density factors are summarized in figure 2.6.1. The degree of circuit integration is dependent upon the number of gates per chip. The following definitions are useful:

- Small Scale Integration SSI = 10 gates/chip
- Medium Scale Integration MSI = 100 gates/chip
- Large Scale Integration LSI = 1000 gates/chip
- Very Large Scale Integration VLSI = 10,000 gates/chip
- Ultra Large Scale Integration ULSI = 100,000 gates/chip
- Super Large Scale Integration SLSI = 1,000,000 gates/chip

Optical lithography is limited to 1 micron or greater. Electron Beam lithography is limited to 0.25 micron or greater. X-ray lithography can be used for even smaller lines, although x-ray lithography is currently being used down to 0.3 microns.

As more gates are added to a VLSI device, the number of pins per chip must increase. The equation for the number of pins required for a ULSI device is indicated at 370 pins. It is possible to reduce the number of pins required on ULSI devices by putting complete systems (or subsystems) on a chip with on-system interconnections, thus reducing the number of external pins needed.

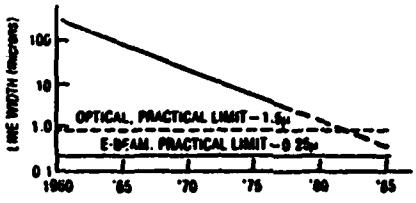
As may be seen in figure 2.6.1, reducing the lithography for a given technology such as silicon MOSFETs reduces the delay-power product. For example, going from the current 3 micron lithography to 0.3 micron lithography reduces the delay power product from 3 picoJoules to 5 femptoJoules: a reduction of 600 to one.

Another factor which must be considered is that the practical limit of dissipation per chip is about 10 Watts, with the preferred level around 2 Watts. This dissipation is limited by the problem of conducting the heat from the chip to maintain the junction temperatures below the limit for the circuits. To cite an example, for a ULSI chip with 10^7 gates, the maximum power delay product with a 1 gHz clock is 10 femptoJoules. Note that this level has been achieved with the Bell Laboratories' 0.3 micron Si MOSFET using x-ray lithography. The relative size of a 0.2 micron line is shown in figure 1.2.6.2. Ultra large scale integration can be achieved by submicron lithography. Hewlett-Packard has already fabricated a 32 bit microprocessor with 450,000 transistors.

Figure 1.2.6.1 COMPONENT DENSITY FACTORS

LITHOGRAPHY METHOD LIMITS LINE WIDTH

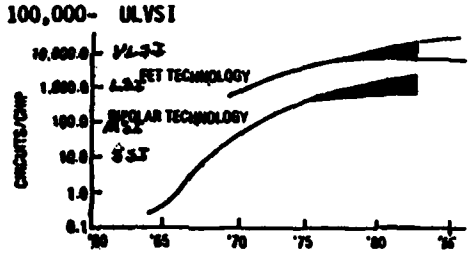
- OPTICAL LITHOGRAPHY GOOD TO 1 μ
  BELOW 1 μ NEED TO USE
  - ELECTRON BEAM LITHOGRAPHY GOOD TO 0.25 μ
  - XRAY LITHOGRAPHY FOR SMALLER LINES



LINE WIDTH LIMITS NO. OF GATES/CHIP

- HAVE ACHIEVED VLSI DEVICES IN SAMPLE QUANTITIES
- AS NO. OF CIRCUITS INCREASE, NO. OF PINS PER CHIP MUST INCREASE

$P = 0.37C^{0.6}$ ⟶
- 40 PINS FOR LSI
- 90 PINS FOR VLSI
- 370 PINS FOR ULSI



POWER DELAY PRODUCT MUST DECREASE AS GATES/CHIP INCREASE

	GATES/CHIP	MAXIMUM PERMISSABLE POWER-DELAY** 1.0 MHz CLOCK	1 gHz CLOCK
ULSI	10^5	10. PJ	.01. PJ
VLSI	10^4	100 PJ	.1 PJ
LSI	10^3	1000 PJ	1. PJ

**MAXIMUM LIMITED BY POWER DISSIP/AREA

CALENDAR TIME INCREASING

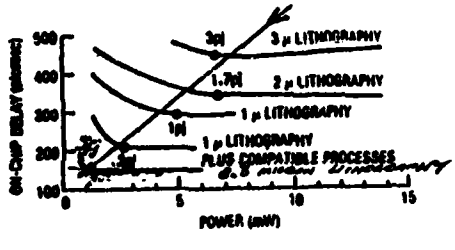


FIGURE 1.2.6.2

VLSI ENABLED BY REDUCTION IN FEATURE SIZE

100,000 GATES/CHIP BY THE 1980's

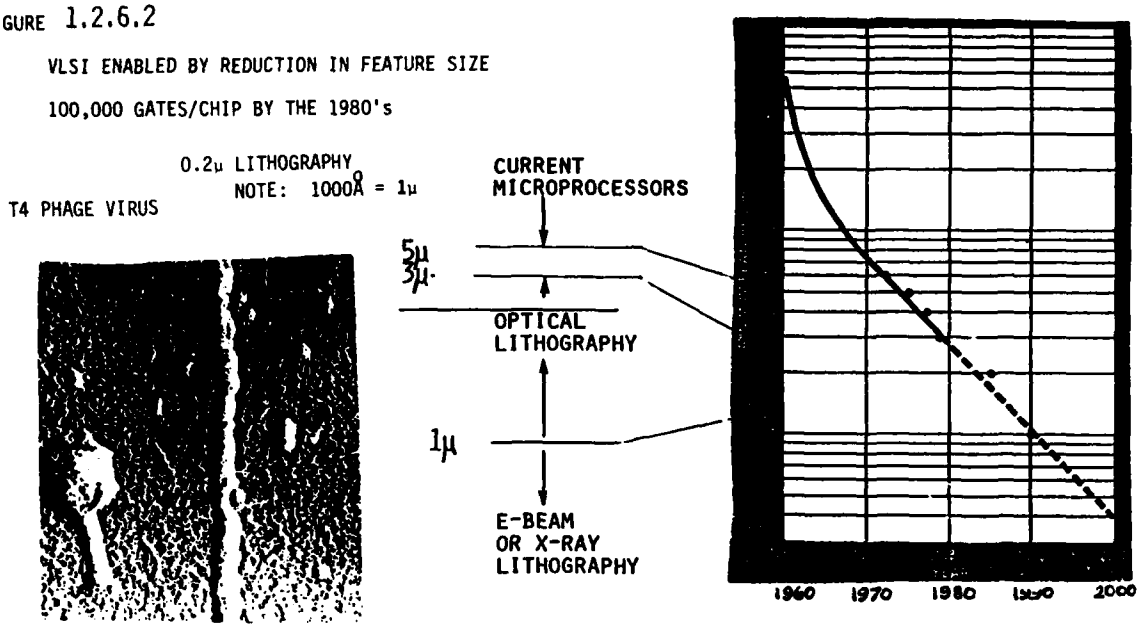
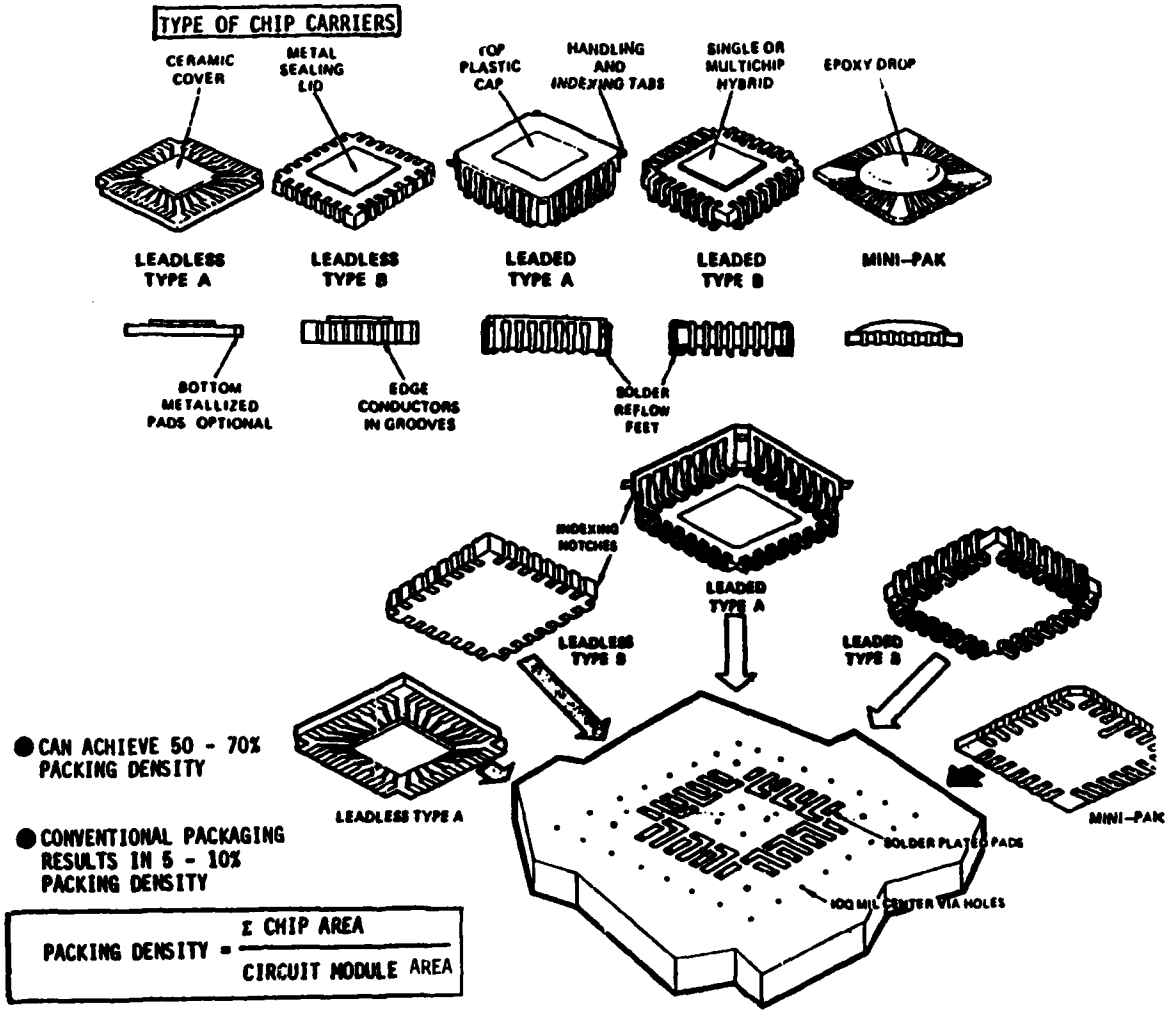


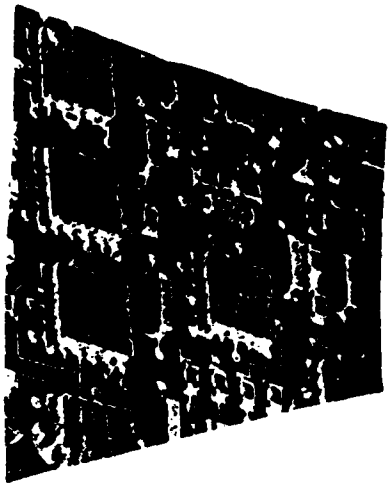
FIGURE 1.2.6.3 HIGH DENSITY PACKAGING



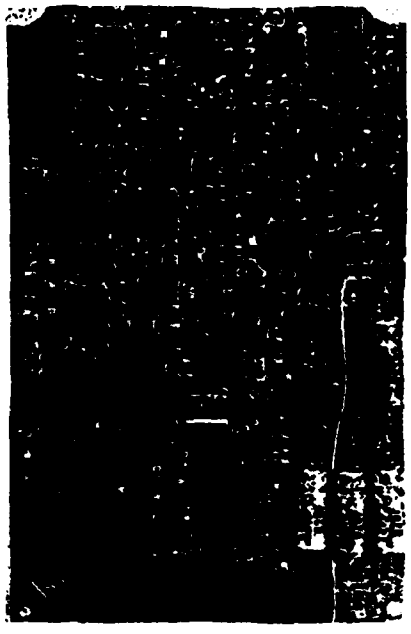
Another factor in achieving packaging density is the number of VLSI chips which can be mounted on a board. The use of leadless chip carriers as illustrated in figure 1.2.6.3 permits an increase in packaging density of 5 to 10 times greater than the use of conventional packaging (i.e. use of chips in a conventional dual in-line package). An sample configuration is shown in figure 1.2.6.4. A method for conducting more heat from a chip with high power dissipation is the use of a beryllia carrier featuring a finned heat sink (see figure 1.2.6.5).

FIGURE 1.2.6.4

LEADLESS CARRIERS INCREASE BOARD DENSITY 6:1

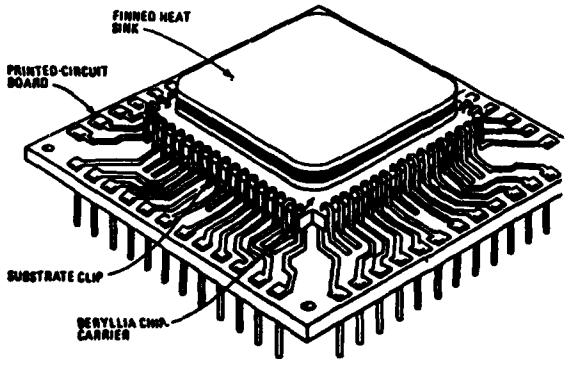


Nested Footprints Accomodate Various Ceramic Leadless Chip Carrier Sizes



- 10 mil Circuit Spacing Minimum
- 10 mil Lines
- 30 mil Nominal Layout Spacing

FIGURE 1.2.6.5 BERYLLIA CARRIER COOLS HOT CHIPS

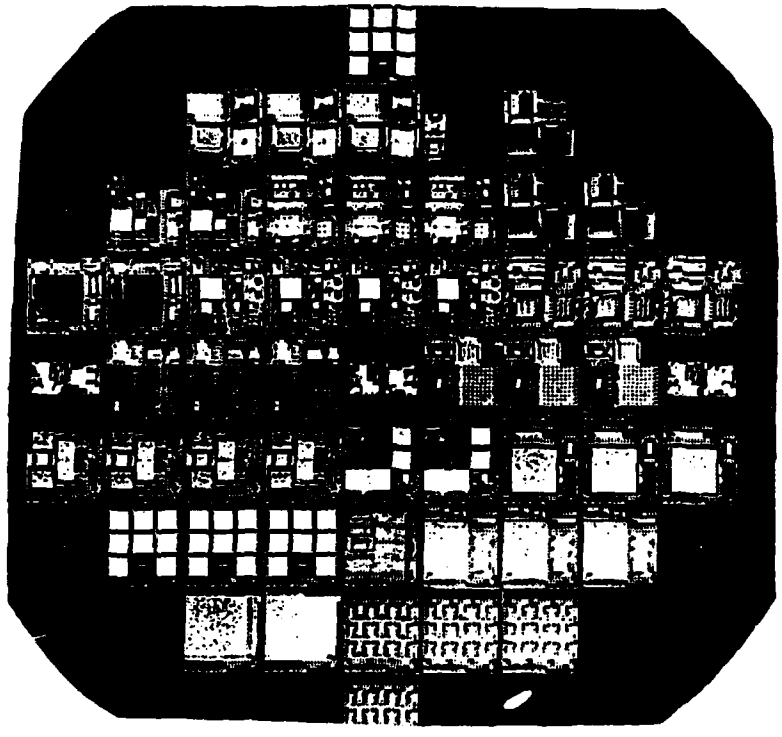


Aids in Increasing Chip Density on Circuit Boards

SPEEDY UNIVAC CARRIER

Another interesting possibility for higher packaging densities is the use of Wafer Scale Integration (WSI). This technique (illustrated in figure 1.2.6.6) uses a single silicon wafer on which a variety of different chip types, including some replicated chips, are diffused into the wafer. The interconnections between the different chips can also be diffused on to the wafer in order to provide a complete system on a wafer.

FIGURE 1.2.6.6 WAFER SCALE INTEGRATION (WSI)



- USC/ISI Designed
- DARPA Funded
- USC/ISI Prepared Masks
- Semiconductor Manufacturer Fabricates
- USC/ISI Uses for Low Cost Custom Chips
- Reeve Peterson at NOSC Sees WSI For Fault Tolerant Systems
- WSI Can Result in System on Wafer with Interconnects Diffused into Wafer

18 DIFFERENT CHIP TYPES — SOME REPLICATED

1.3.0 MICROPROCESSOR FAMILIES

1.3.1 Introduction

The microprocessor revolution was launched in 1972 by the Intel 4004, followed by the Rockwell PPS-4, which were 4 bit PMOS technology microprocessors. In the intervening 9 years there has been a virtual revolution in microprocessor types, with each generation increasing in capability and throughput. The current generation of 16 bit and 32 bit microprocessors are challenging minicomputers and even mainframe computers such as the IBM 370 for throughput. Projections hold that microprocessors will exceed the throughput of the current super computers such as the Cray-I and the Amdahl-470 before the end of the decade.

Mr. Louis J. Urban, Chairman of Working Group 06, provided the following statistics quantifying the plethora of microprocessor types taken from his UCLA lectures on Avionics. These statistics are summarized in table 1.3.1.1 below:

Table 1-3.1-1 MICROPROCESSOR STATISTICS

■ The attributes of the microprocessor are shown on the left, the number of microprocessors having that attribute is shown on the right.

Microprocessor Types	
Major Types	125
Variations	242
No. of Manufacturers	44
Size of Arithmetic Logic Unit	
4 bit	123
8 bit	98
16 bit	33
32 bit	4
Other Length (1,2,12 bit)	8
Type of Circuit Technology	
NMOS	108
PMOS	66
CMOS	37
Bi-POLAR	28
No. of Instructions	
1-39 Instructions	26
40-59 Instructions	58
60-79 Instructions	53
80-99 Instructions	25
100-139 Instructions	15
more than 140 Instructions	11

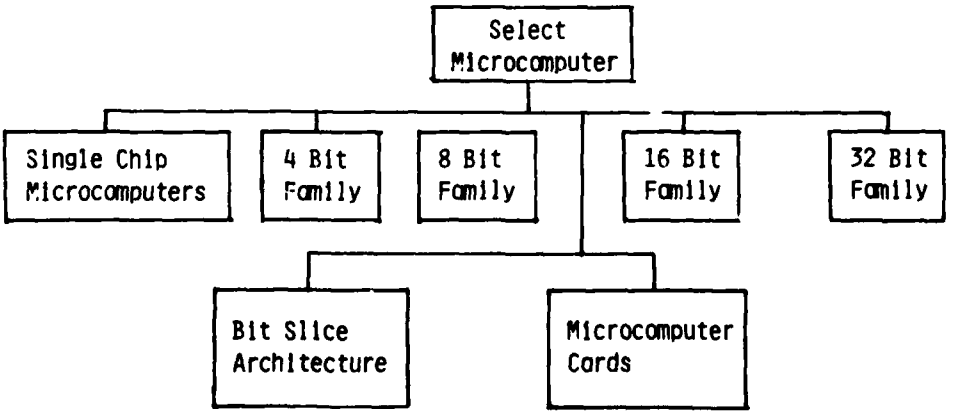
The 4 bit microprocessor is still widely used in dedicated controllers. The use of 4 bit microprocessors in calculators and automobiles have made this type the most widely used.

The 8 bit microprocessors are the workhorse of dedicated micro-computer applications. The use of distributed architecture plus sophisticated peripheral chips have provided great capability for the 8 bit microprocessors.

The newer 16 bit and new 32 bit microprocessors provide the capability of minicomputers and main frame computer CPU's on a single chip.

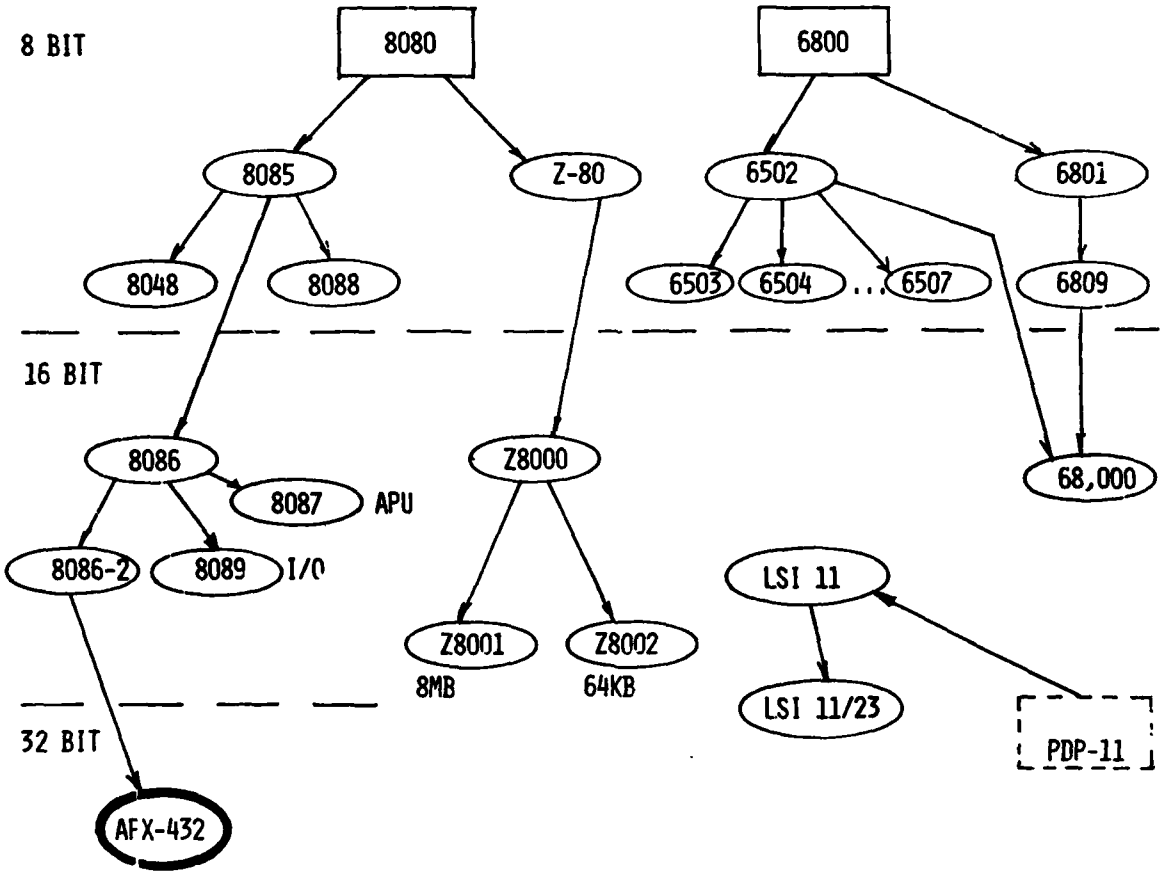
The current microcomputer families are shown in figure 1.3.1.1. Another popular type of microprocessor is the single chip microcomputer which includes the central processing unit (CPU), random access memory, read only memory (or programmable ROM), digital I/O (serial and parallel), and in some cases analog-to-digital and digital-to-analog conversion on the single chip. The single chip microcomputers are utilized for dedicated controllers and smart peripheral chips.

FIGURE 1.3.1.1 CURRENT MICROCOMPUTER FAMILIES



Even though a large number of microprocessors have been developed, the natural selection process of the market place has made only a small number of the microprocessors dominant. These dominant family trees (figure 1.3.1.2) propagate into new generations which are related to the original parent product. The Intel 8080 and the Motorola 6800 have become the dominant 8 bit microprocessors. The 8080 is descended from the original Intel 4004 (later improved to the 4040). The Rockwell PPS-8, which descended from the PPS-4, did not survive in the market place. The 6502 which originally was an improved pin-for-pin replacement for the 6800 design by MOS Technology; however, a law suit by Motorola against MOS Technology resulted in a design change that made the 6502 slightly incompatible to the 6800. Rockwell second-sourced the 6502 and used it in its AIM microcomputer; Apple used the 6502 as the CPU for the Apple Computer. As a result, the 6502 is one of the most widely used 8 bit microprocessors available today.

FIGURE 1.3.1.2 DOMINANT MICROPROCESSOR FAMILY TREES



The very popular 8080 was the basis of the first home microcomputer which spawned the S-100 bus phenomena now widely used in the hobby market as well as many small business computers. The 8080 descendents include the Intel 8085 and the Zilog Z-80, which both execute the 8080 instruction set as a subset which has become a de facto standard.

The three most popular and now mature 16 bit microprocessors, derived from the 8080, 6800, and Z-80 8 bit microprocessors, are the Intel 8086, the Motorola 68000, and the Zilog Z8000. The first of the new commercially available 32 bit microprocessors is the Intel iAPX-432, which is a descendent of the 8080, but with a radical change in architecture.

Another development is the descendent of microprocessors from the popular 16 bit minicomputers. An example is the descendent of the Digital Equipment Corporation PDP-11 minicomputer: the LSI-11, a 16 bit microprocessor which executes the DEC PDP-11 assembly code. This strategy on the part of minicomputer manufacturers such as DEC captures the vast amount of minicomputer software already in existence for a new microprocessor which emulates the minicomputer CPU.

Another type of microprocessor - incorporating bit-slice architecture - has been used to emulate existing computers in order to capture existing software. The AmD 2901 4-bit-slice family of chips has been widely used to emulate existing computers. The 2901 can be microcoded to emulate a given computer instruction set. The draw-back of the bit-slice microcomputer is the relatively large number of chips required to implement a CPU. Although bit-slice architecture has been important in the past, it will be less important in the future as the trend towards high-level language and extremely powerful microprocessors continue. If an organization's software is in high level language (HLL), then only a compiler is needed to capture software for a new microprocessor

1.3.2 New 32 Bit Microprocessors

The advent of very large scale integration (VLSI) with 1 to 3 micron lithography has enabled the development of powerful new 32 bit microprocessors as shown in figure 1.3.2.1. The three recently announced 32 bit microprocessors will be described below: the Intel iAPX-432, the Bell MAC/32, and the HP 32 bit microprocessor. Only the iAPX-432 will be commercially available; the other two are reserved for internal company products.

FIGURE 1.3.2.1 PRIMARY TECHNOLOGY THRUSTS

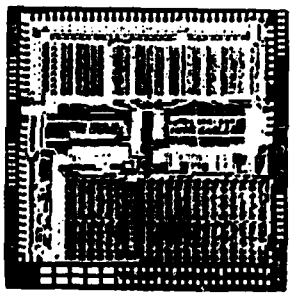
VLSI WITH 1-3 MICRON FEATURE SIZE HAS ENABLED POWERFUL NEW 32 BIT MICROPROCESSORS

INTEL iAPX43201
100,000 DEVICES

3 CHIP SET
- FASTER THAN IBM 370/148
- ADA STATEMENTS COMPILE TO SINGLE INSTRUCTION
- 80 BIT FLOATING POINT MULTIPLY (26μS)

BELL MAC/32
100,000 DEVICES

HP 32 BIT MICROPROCESSOR
450,000 DEVICES



1.5 MICRON LINES
ELECTRON BEAM LITHOGRAPHY

The iAPX-432 32 bit microprocessor shown in figure 1.3.2.2 consists of a 3 chip set: the 43201 instruction decoder, the 43202 micro execution unit, and the 43203 input/output support chip. The 432 CPU has a total of 225,000 transistors on the three chips. The 432 has one of the first Ada compilers and its architecture was specifically designed with Ada in mind. A two phase 8MHz clock provides a microcycle of 62.5 nonoseconds.

The throughput of the 432 (see figure 1.3.2.3) is enhanced by software-transparent multi-processing. This feature permits two 432 CPU's to exceed the throughput (1 MOP/second) of a VAX 780 and five 432 CPU's to exceed the throughput (2 MOP/second) of an IBM 370/158. The three chip set costs $3000 with deliveries scheduled for late 1981, but Intel projects the cost to be $200 by 1984. The 432 has a microcoded operating system which is 20-30 times faster than a mainframe software operating system.

FIGURE 1.3.2.2 INTEL iAPX 432 3 CHIP-SET MICROPROCESSOR

43201 INSTRUCTION DECODER

100,000 DEVICES

NEW ADA 32 BIT

225,000 TRANSISTORS

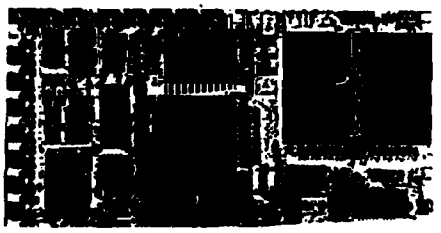
64 PIN QUIP / 5 VOLT PS / 2.5 WATTS EACH

- ADA – THE 432'S NATIVE LANGUAGE
- iMAX OPERATING SYSTEM
- ONE OF THE FIRST ADA COMPILERS AVAILABLE
- 2 PHASE 8 MHZ CLOCK

43202 MICROEXECUTION UNIT

60,000 DEVICES

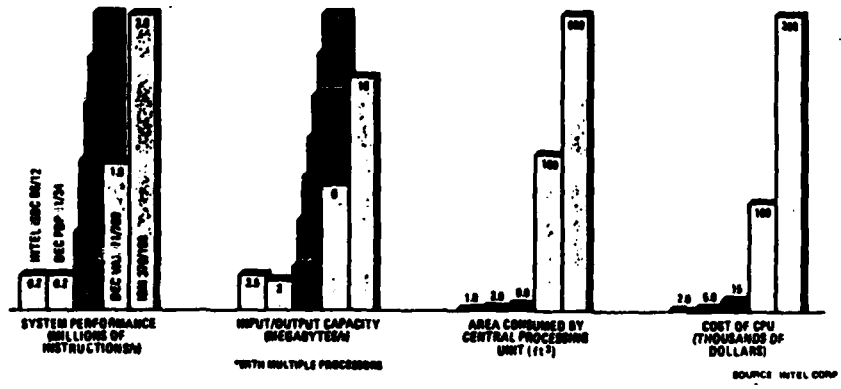
43203 INPUT/OUTPUT SUPPORT



65,000 DEVICES

FIGURE 1.3.2.3 iAPX 432 WITH SOFTWARE-TRANSPARENT MULTIPROCESSING

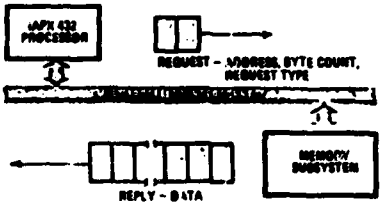
THRU-PUT INCREASED BY ADDING 432 PROCESSORS



COMPARISON

8086
PDP 11/32
iAPX 432
VAX 11/780
IBM 370/158

AVAILABLE 1982 — $3,000/CHIP SET ➡ $200 BY 1984



- MICROCODED OPERATING SYSTEM EXECUTES SEND MESSAGE 20-30 TIMES FASTER THAN MAINFRAME SOFTWARE O/S
- QUANTITIZED MEMORY REQUESTS
- VIRTUAL MEMORY ADDRESSES - 2^{40} BYTES MAX (1000 GIGABYTES)

FIGURE 1.3.2.4 iAPX 432 CHARACTERISTICS

EXECUTION TIMES

Functional unit	Typical operation	Execution time at 8 MHz (μs)
Variable-precision integer arithmetic unit	32-bit integer multiply	8.25 (16 μs on IBM 370/148)
Microprogrammed floating point arithmetic unit	80-bit floating multiply	26.125 (28.5 μs on IBM 370/148)
Barrel-shift unit	32-bit field extract	1.875
Address generator with associative cache of least recently used addresses	32-bit memory access	0.75
Silicon operating system	send message	80.875

FASTER THAN IBM 370/148

MOST OF MICROCODE ALLOCATED TO O/S

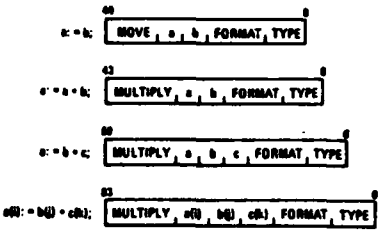
Function	Bits	Percentage
Basic instruction set	3,680	6
Floating-point arithmetic	11,680	18
Run-time environment	6,400	10
Virtual addressing	4,800	7
Fault handling	2,840	4
Silicon operating system	26,400	40
Multiprocessor control	8,840	13
Debug services	1,280	2
Total	64-K	100

OBJECT ORIENTED ARCHITECTURE ➡ OBJECT ADDRESSING

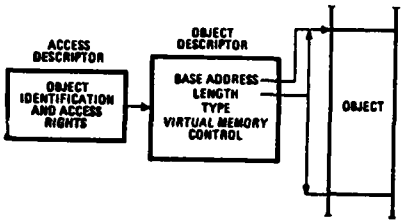
- DATA OBJECTS
- SYSTEM OBJECTS (HARDWARE - RECOGNIZED)
  - DOMAIN OBJECTS - ADDRESS ENVIRONMENT OF PROGRAM MODULE
    - ACCESS DESCRIPTION
    - LINKS TO OTHER DOMAINS
    - PROGRAM MODULE'S INSTRUCTION AND DATA OBJECTS
    - PUBLIC AND PRIVATE PARTS (PRIVATE NOT ACCESSABLE)
  - CONTEXT OBJECTS - SUPPORT DYNAMIC ALLOCATION OF MEMORY
    - SUPPORT SHARED, RECURSIVE, REENTRANT PROCEDURES
    - PROVIDE EACH PROCEDURE ACTIVATED WITH DATA OBJECT
  - PROCESS OBJECT - INDEPENDENT CONCURRENT TASK
  - STORAGE RESOURCE OBJECT - PORTION OF FREE STORAGE

FIGURE 1.3.2.5 iAPX 432 CHARACTERISTICS

VARIABLE SIZE INSTRUCTIONS NOT CONSTRAINED TO BYTE BOUNDARIES



MEMORY REQUESTS VIA TWO LEVEL OPERATION TO ACCESS GIVEN OBJECT



ADA IS 432 IMPLEMENTATION LANGUAGE

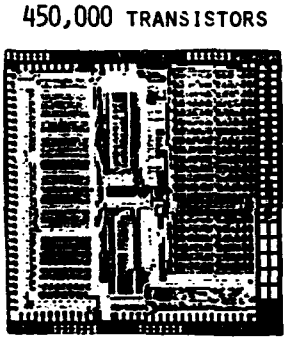
- GOAL OF ADA & 432 - INCREASE PROGRAMMER PRODUCTIVITY
  - INCREASE SOFTWARE RELIABILITY
  - LOWER SOFTWARE LIFE CYCLE COSTS
- ADA INSPIRED BY PASCAL - SUPPORTS LARGE SCALE, MODULAR SOFTWARE
- ADA PACKAGE CONSTRUCTS - CONSISTENT WITH 432 OBJECT ORIENTED MODULARIZATION
  - PACKAGE RESTRAINTS ACCESS TO OBJECT
  - PACKAGE DEFINES OBJECT AND OPERATIONS
- ADA PERMITS SPECIAL MACHINE ACCESS PACKAGE - ELIMINATES NEED FOR 432 ASSEMBLER
- ADA USED TO WRITE iMAX EXECUTIVE
- ADA IS BASIS FOR 432's INTEGRATED PROGRAMMING SYSTEM
  - SEPARATE COMPILATION IN ADA
  - DEBUGS SOURCE-LEVEL PROGRAMS IN ADA, NOT 432 MACHINE LANGUAGE

The execution time of the 432 (see figure 1.3.2.4) for an 80 bit floating-point multiplication is 26 microseconds, compared with 38 microseconds for the IBM 370/148. Most of the microcode for the 432 (40%) is allocated to the silicon operating system - only 6% is allocated to the basic instruction set. Other microcoded functions include floating-point arithmetic, multi-processor control, virtual addressing (a terabyte, or 10^{12} bytes), and the run-time environment. The architecture is object oriented like the new DoD high-level language Ada, which is the native language of the 432. The Ada compiler for the 432 compiles into a single instruction for each line of Ada source code.

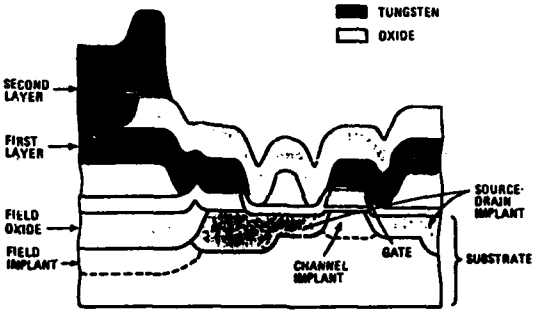
Other characteristics of the 432 are portrayed in figure 1.3.2.5. One unique characteristic of the 432 is the use of variable instruction size not constrained to byte boundaries. The 432 does not have machine assembly language accessible by the user. The integrated programming system compiles in Ada and debugs source programs in Ada.

The new HP 32 bit microprocessor has achieved the highest densty thus far with 450,000 transistors on a single chip. The characteristics of the HP 32 bit microprocessor are summarized in figure 1.3.2.6. The circuit technology is NMOS with 1.5 micron line width and 1 micron line spacing. Pipelining overlays three instructions to achieve a 55 nanosecond microcycle with an 18MHz clock. The throughput is fast with a 64 bit floating point multiplication in 10.4 microseconds and a 32x32 bit fixed point multiplication in 1.8 microseconds. An extensive instruction set utilizes 9000 38 bit microcode words.

FIGURE 1.3.2.6 NEW HEWLETT PACKARD 32 BIT MICROPROCESSOR



18 MHZ CLOCK
➡N-MOS 3
1.5μ LINE WIDTH
1μ SPACES
8 MASK PROCESS



- PIPELINING OVERLAYS THREE INSTRUCTIONS FOR 55 NANOSEC MICROINSTRUCTION
- FAST PROCESSING
  - 32 x 32 BIT MULTIPLY 1.8μsec
  - 64 x 32 BIT DIVISION 3.5μsec
  - 64 BIT FLOATING POINT MULT (IEE FP STD) IN 10.4μsec
- EXTENSIVE 32 BIT INSTRUCTION SET UTILIZES 9000 38 BIT MICROCODE WORDS
- REGISTER BASED ARCHITECTURE WITH N-BIT SHIFT REGISTER (BARREL SHIFTER)
  - FOUR SEPARATE STACK PRINTERS
  - 28 x 32 BIT FILE
- INCLUDES DEBUGSING INSTRUMENTS
  - BREAKPOINTING
  - SINGLE STEPPING
  - TRACING

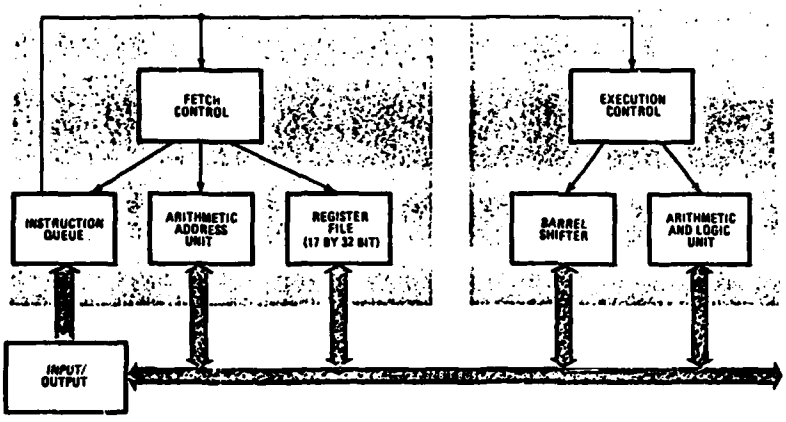
NO PLANS TO MARKET ANNOUNCED - PROBABLY FOR HP PRODUCTS

Bell Laboratories' new MAC/32 32 bit microprocessor is shown in figure 1.3.2.7. The MAC/32 utilizes CMOS technology with 3.5 micron lithography with 100,000 transistors in the CPU. The MAC/32 uses a 32 MHz clock for a 31 nanosecond microcycle. The register based architecture uses 17x32 bit files and uses separate units for instruction fetch and execution.

Another famous 32 bit mainframe - the IBM 370 - has been reduced to a VLSI CPU chip as shown in figure 1.3.2.8. IBM fabricated this 5000 gate chip using computer aided design (CAD) techniques in 9 months. The CPU utilizes bi-polar silicon technology on a 7mm x 7mm chip with a total of 45,000 components on the chip.

FIGURE 1.3.2.7 NEW BELL LABS MAC-32

32 BIT C-MOS MICROPROCESSOR



100,000 TRANSISTORS

TWIN TUB C-MOS TECHNOLOGY

EXTREMELY LOW POWER

3.5 MICRON LITHOGRAPHY

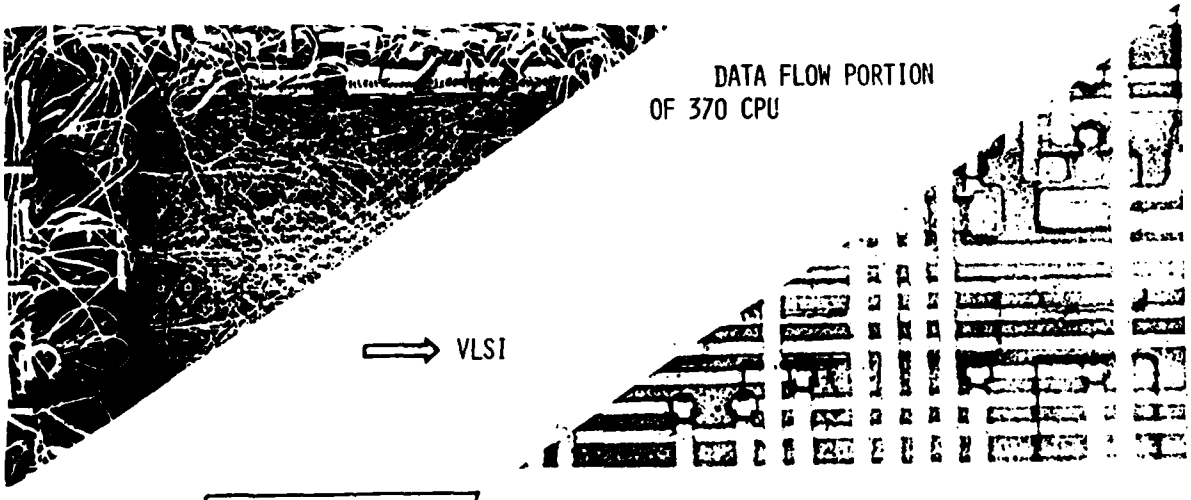
2.5 MICRON PLANNER (9 x 9 CHIP SIZE)

32 MHZ CLOCK

- REGISTER BASED ARCHITECTURE - 17 x 32 BIT FILE
- SEPARATE UNITS FOR INSTRUCTION FETCH AND EXECUTING
- INSTRUCTION QUEUE
- HIERARCHY OF 4 PRIVLEDGE LEVELS - ISOLATES SYSTEM AND USER SOFTWARE

NO PLANS TO MARKET. MAC-32 WILL BE USED ON WESTERN *ELECTRIC EQUIPMENT*

FIGURE 1.3.2.8 IBM FABRICATES VLSI VERSION OF 370 CPU



9 MONTH DESIGN TIME USING CAD

- 7MM x 7MM CHIP
- 5000 GATES (45,000 COMPONENTS)
- BI POLAR SILICON

1.3.3 New 16 Bit Microprocessors

The last year (September 1980 to September 1981) has seen a number of new VLSI 16 bit microprocessors joining the mature 16 bit microprocessors, summarized in figure 1.3.3.1. The attributes listed in the figure emphasize that the new microprocessors feature the use of virtual memory management to expand the memory address space.

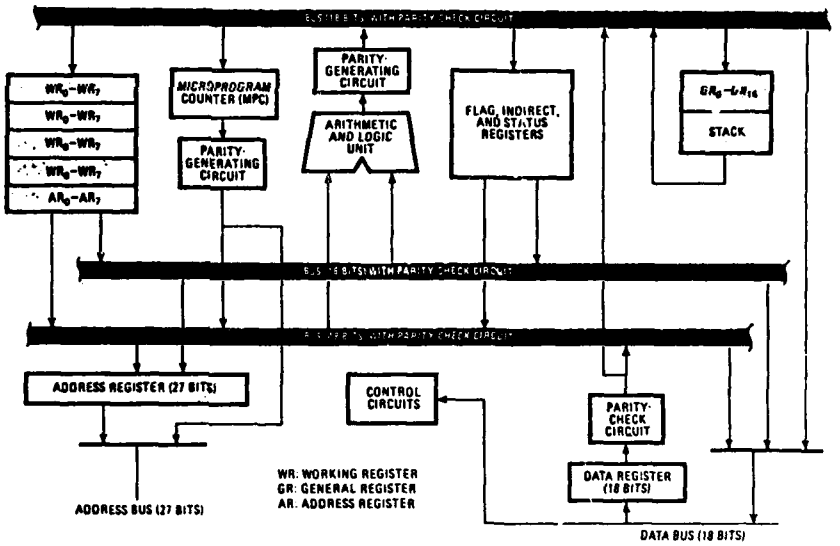
Figure I-3.3-1 VLSI ENHANCES NEW 16 BIT MICROPROCESSORS

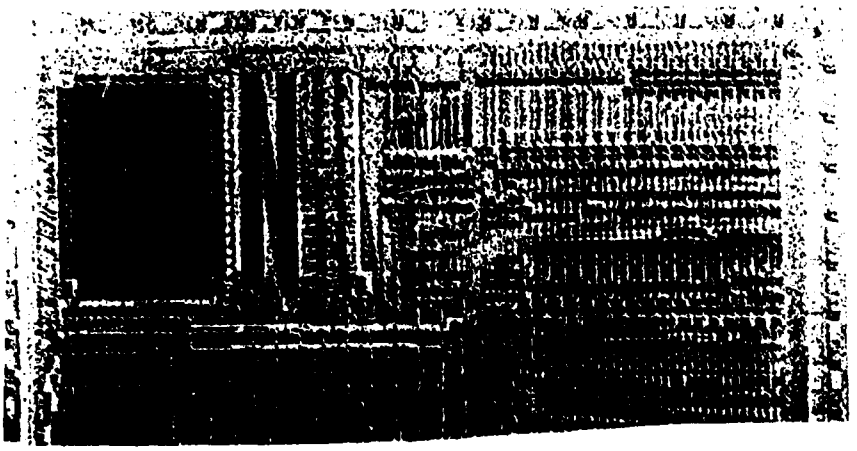
	μPROGRAMMED	VIRTUAL MEMORY	PIPELINED	MULTIPLIER	BARREL SHIFTER	SELF TEST	REGISTER BASED	MEMORY BASED
• NEW 16 BIT MICROPROCESSORS								
Intel iAPX 286	•	•	•	•			•	
TI 99000	•		•			•		•
Toshiba CMOS/SOS	•		•				•	
National 16,000	•	•	•				•	
Phillips Gloeilampen Fabriken	•	•						
• JOIN THE MATURING 16 BIT MICROPROCESSORS								
Intel 8086	•		•	•			•	
Motorola 68000	•						•	
Zilog Z8000			•				•	

The Fujitsu FSSP 16 bit CMOS microprocessor announced in 1981 is shown in figure 1.3.3.2. The FSSP is a 10,000 gate VLSI chip with 16M bytes of addressable memory space. The CMOS technology assures a low power dissipation of 130 mW when operated at 2.5MHz with a single 5 volt supply voltage. The FSSP has 117 micro instructions and features on chip parity checking.

FIGURE 1.3.3.2

- 10,000 Gate Chip
- 16 Mbyte Addressable Memory
- 2.5 MHz Clock/ 130 MW @ 5V
- 117 Micro-instructions

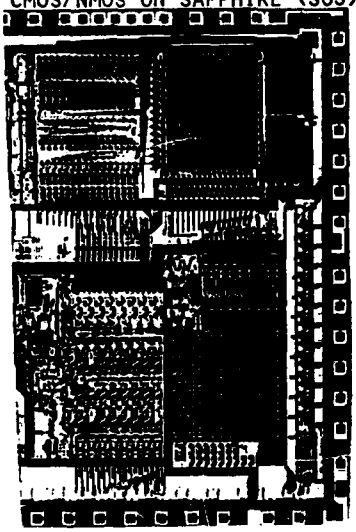


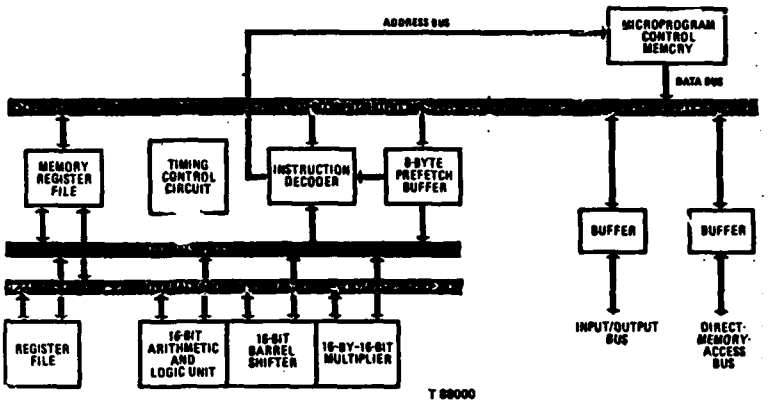


The Toshiba 88,000 new CMOS/NMOS 16 bit microprocessor (figure 1.3.3.3) uses 3.5 micron lithography to achieve 10,000 gates on the CPU chip, equally divided between NMOS and CMOS gates. The CMOS/SOS technology reduces the CPU dissipation to 700mW compared with 3W for an all-NMOS silicon chip. This unit has 3-6 times the throughput of an 8086 or Z8000 16 bit microprocessor. The gate delay of 700 picoseconds contributes to fast processing: for example, a 16x16 bit floating point multiplication is executed in 1.6 microseconds.

FIGURE 1.3.3.3

CMOS/NMOS ON SAPPHIRE (SOS)





- Barrel shifter — 1-15 shifts in 400 ns
- 12,000 gates equal divided among CMOS and NMOS GATES
- 700 MW (CF 3W for Si)
- Gate delay of 700 picosec
- Hardware floating point and decimal processing
- 151 instructions / 300 ns microcycle
- Microprogram in off board ROM
- 3.5 micron lithography

SOS SPEED EQUIV. TO 2 MICRON Si

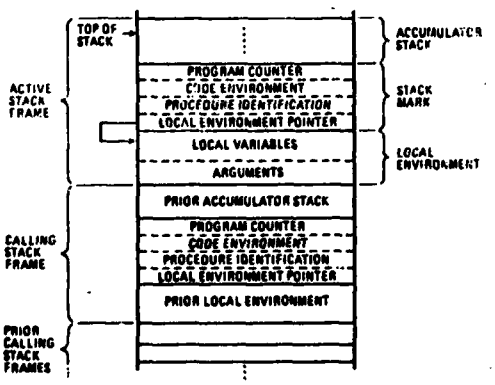
- 3-6 times faster than 8086 or Z8000
- 16 x 16 bit mult. 1.6 microsec, fixed point

The new Rockwell AAMP 16 bit microprocessor is illustrated in figure 1.3.3.4 which has 68,000 devices on a single 200 x 200 mil chip. The AAMP uses CMOS/SOS technology with 2 micron lithography to achieve high throughput (0.3 MOPS/second with a 12% floating point mix) and a low power dissipation (400mW) with a single 5V power supply voltage. The unit uses a 20MHz clock which results in a 32-bit multiply in 15 microseconds. The AAMP was designed for avionics applications and will be sampled in late 1981.

FIGURE 1.3.3.4 ROCKWELL AAMP NEW 16 BIT MICROPROCESSOR

CMOS ON SAPPHIRE

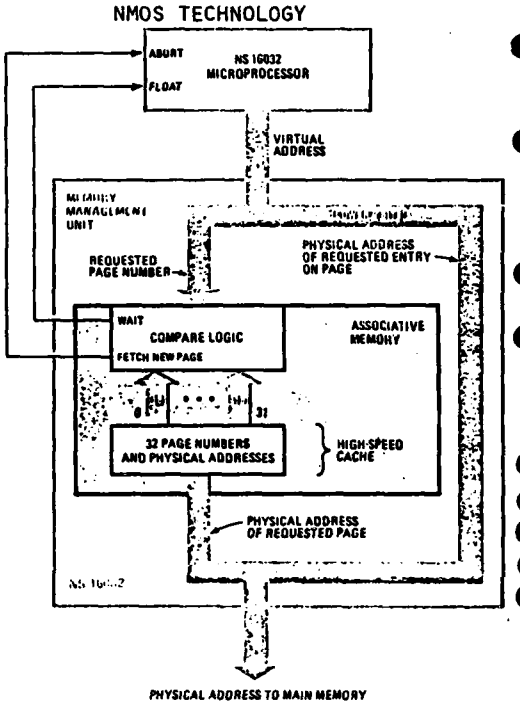
2 MICRON LITHOGRAPHY



- 0.4 Watts / 5 Volt Supply
- Advanced Architecture Microprocessor (AAMP)
- Designed for Avionics Applications
- Stack Intensive Architecture
- 1K x 40 bit Microcode
- 16 bit Mult. 4.85 Microsec
- 32 bit mult. 15.05 Microsec
- Addresses 16 Mbyte Memory Space
- Stack Oriented Architecture Enables Multi-tasking
- 68,000 Devices
- 214 x 216 Mil Chip
- 20 MHz Clock
- 300,000 Instructions/Sec (12% F.P. Mix)

The new National Semiconductor NS-16,000 16-bit microprocessor (figure 1.3.3.5) has 60,000 transistors on a 330 mil square chip, which will address 16 Mbytes of memory using the companion 16032 memory management unit. The 16,000 has a 100 nanosecond microcycle and features a 32 bit internal data bus chip self-test.

FIGURE 1.3.3.5 NATIONAL SEMICONDUCTOR'S NEW NS 16,000 16 BIT MICROPROCESSOR



- MICROCODED INSTRUCTIONS
  - 1300 x 8 BITS MICROCODE MEMORY
  - 100 NS MICROPROGRAM CYCLE
- VIRTUAL MEMORY MANAGEMENT
  - 16 MBYTES ADDRESS SPACE
  - SEMICONDUCTOR MEMORY/MASS MEMORY TRANSPARENT TO PROGRAMMER
- SEPARATE FETCH AND EXECUTION UNITS
  - SPEEDS THRUPUT
- PIPELINES EXECUTION
  - INSTRUCTION LOADING
  - DECODING
  - EXECUTION
- 32 BIT INTERNAL DATA BUS
- 16 BIT MEMORY MEMORY INTERFACE
- 60,000 TRANSISTORS
- 8.4 x 8.4 MM CHIP
- INCLUDES 127 BYTES SELF TEST

NS 16032 MICROPROCESSOR
NS 16082 MEMORY MANAGEMENT UNIT

1.3.4 Mature 16 Bit Microprocessors

The following section describes three 16 bit microprocessors introduced just a few years ago, which are now considered mature due to the variety of support peripheral chips and support software, including compilers for high-level languages - the Intel 8086, the Motorola 68000, and the Zilog Z8000 - now available.

The Intel 8086 (figure 1.3.4.1) central processor unit has two co-processors to enhance the throughput which can be obtained when this family of 16 bit microprocessors is combined in a system. The 8087 floating point arithmetic processor (64 bit arithmetic) and the 8089 I/O processor provide multiple parallel processing. The 8086 microprocessor is seven to twelve times faster than the 8080. The 8086 and 8089 are both connected to the local 16 bit bus with the 8089 IOP providing access to the I/O devices which are not accessible to the 8086. The 8086 provides a number of simultanious processing functions such as instruction execution at the same time it is fetching the next instruction.

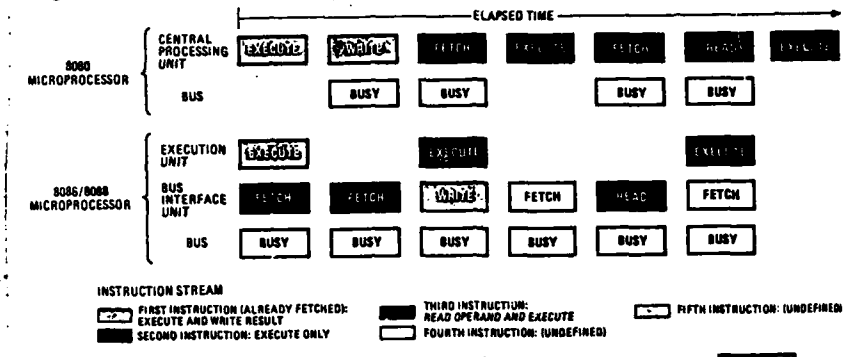
The 8087 FPP performs 64 bit FP multiplications in 18 microseconds and calculates a tangent in 110 microseconds. The 8087 and 8086 are both connected to the local instruction bus and decide which instruction type should be executed. For example, if the instruction is an arithmetic calculation, then the 8087 executes the instruction; otherwise the 8086 executes it. Intel has recently introduced the iAPX-286 processor which is some 5 times faster than the 8086 and is also downward compatible.

The Motorola 68000 (figure 1.3.4.3) is the most powerful of the mature 16 bit microprocessors, addressing 16M Bytes of memory compared with 1M Byte for the 8086. The 68000 operates at a clock speed of 8MHz and is some 10 to 25 times faster than the 6800. The chip has 68,000 transistors (13,000 gates) using 3 micron lithography. The 68000 has a large complement of peripheral chips. Although the 68000 is compatible with the 8 bit 6800/6502 bus, it also has a 16 bit versabus to take full advantage of its 16 bit architecture.

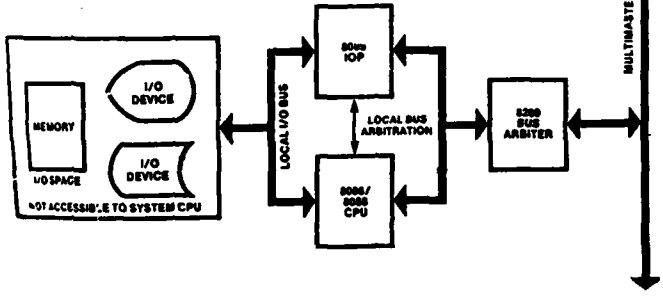
MATURE

FIGURE 1.3.4.1 THE ~~NEW~~ 16 BIT MICROPROCESSORS

- INTEL 8086 CPU
  - CO-PROCESSORS -8087 FLOATING POINT ARITHMETIC PROCESSOR (64 BIT)
    -8089 INPUT/OUTPUT PROCESSOR
  - 7-12 TIMES FASTER THAN 8080
  - ADDRESS SPACE 1M BYTE
  - MULTIPLE PARALLEL PROCESSING



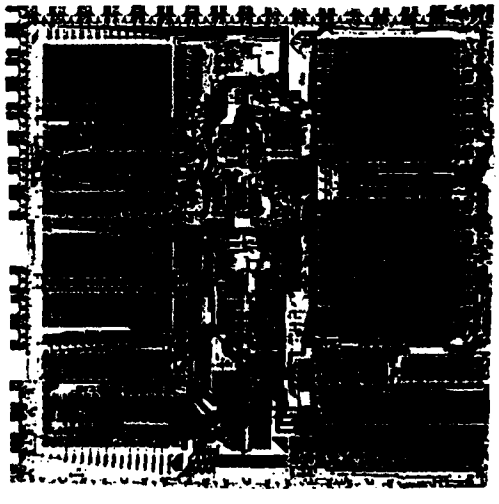
- 8089 I/O COPROCESSOR
- COMPATIBLE WITH MULTIBUS



MATURE

FIGURE 1.3.4.3 THE ~~NEW~~ 16 BIT MICROPROCESSORS (CONT.)

- MOTOROLA 68,000
  - -10-25 TIMES FASTER THAN 6800
  - -281 x 281 MIL
  - -68,000 TRANSISTORS (13,000 GATES)
  - -8 MHz
  - -1.2 w POWER
  - -3.2 MICRON LITHOGRAPHY
  - -ADDRESS 16M BYTES
- 68,000 SUPPORT CHIPS
  - -68120 INTELLIGENT PERIPHERAL CONTROLLER
  - -68122 CLUSTER TERMINAL CONTROLLER
  - -68540 ERROR DETECTION AND CORRECTION UNIT
  - -68451 MEMORY MANAGEMENT UNIT
  - -68450 DIRECT MEMORY ACCESS CONTROLLER
  - -68454 HARD DISK CONTROLLER
  - -68230 PARALLEL INTERFACE UNIT AND TIMER
  - -68341 FLOATING POINT READ ONLY MEMORY
  - -68561 MULTIPROTOCOL COMMUNICATIONS CONTROLLER
  - -68340 DUAL PORT RANDOM ACCESS MEMORY
  - -68453 BUBBLE MEMORY CONTROLLER
  - -68540 SERIAL DMA PROCESSOR
- COMPATIBLE WITH 6800/6500 BUS



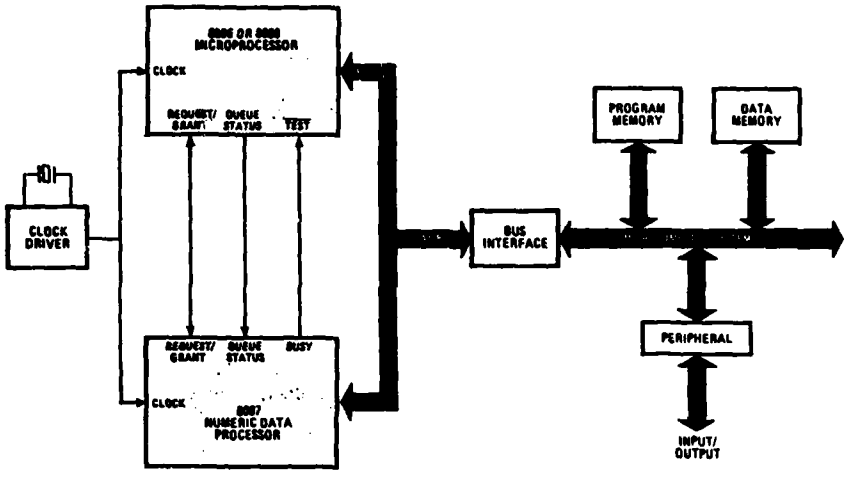


FIGURE 1.3.4.2

MATURE

THE ~~NEW~~ 16 BIT MICRO-PROCESSORS (CONT.)

8087 HI SPEED MATH CO PROCESSOR

COPROCESSOR EXAMINE LOCAL BUS FOR INSTRUCTIONS

330 x 330 MILS
65,000 DEVICES
40 16 BIT REGISTERS

Instruction	Approximate execution time (μs)	
	8087 (5-MHz clock)	8086 emulation
Add or subtract magnitude	14/18	1,600
Multiply (single precision)	18	1,600
Multiply (double precision)	27	2,100
Divide	39	3,200
Compare	10	1,300
Load (single precision)	9	1,700
Store (single precision)	17	1,200
Calculate square root	36	19,600
Calculate tangent	110	13,000
Raise to the appropriate power	130	17,100

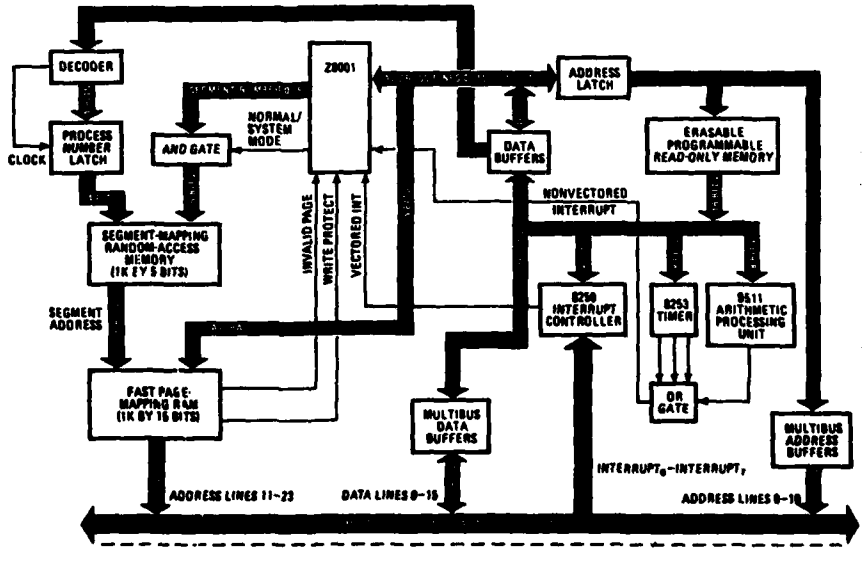
	Range	Precision
Word integer	10^4	16 bits
Short integer	10^9	32 bits
Long integer	10^{18}	64 bits
Packed binary-coded decimal	10^{18}	18 digits
Short real	$10^{\pm 38}$	24 bits
Long real	$10^{\pm 308}$	53 bits
Temporary real	$10^{\pm 4,932}$	64 bits

MATURE

FIGURE 1.3.4.4 THE ~~NEW~~ 16 BIT MICROPROCESSORS (CONT.)

MICROCOMPUTER SYSTEM

- ZILOG Z-8000
  - -ADDRESS 8M BYTES
  - -4MHz CLOCK



The Zilog Z8000 operates with a 4MHz clock and will address 8M bytes of memory space. The Z8000 is not microcoded as is the 68000 and 8086. Consequently, the instruction set cannot be modified in the future. The Z8000 has a Jovial compiler being developed by GD Electronics. This provides a powerful HLL for the Z8000 for USAF applications requiring Jovial (MIL-STD-1789).

1.3.5 Dominant 8 Bit Microprocessors

The 8 bit microprocessor is widely used with the most popular families shown in figure 1.3.5.1. The original 8080 has been widely used in guidance and control applications. For example, the airborne computer in the NASA Dryden supersonic HiMAT research RPV uses two 8080 microprocessors for flight control and guidance. The flight control system includes 3-axis control plus propulsion control. The 8085 and 8088 are enhancements of the 8080 having higher throughput.

The Motorola 6800 and the enhanced 6809 microprocessor have been applied to a variety of systems. The 6800 was one of the original 8 bit microprocessors which became very popular, particularly for communications applictions. A variant of the 6800 is the 6500 microprocessor, originally developed by MOS Technology, second-sourced by Rockwell. Rockwell became the largest supplier of the 6500, primarily because of the popularity of its low cost AIM computer which sells for $175 including microcomputer, keyboard, and display.

The Zilog Z80 captured much of the 8080 market by providing a faster processor with an enhanced instruction set which will execute the 8080 subset.

All of the popular 8 bit microprocessors have had their life extended and their range of applications expanded by a large variety of peripheral chips and mature support software. Virtual memory management chips are extending the addressability range of the 8 bit microprocessors.

Another trend for the 9 bit microprocessors is the conversion of the popular units to CMOS versions to enhance performance and reduce power dissipation. A number of CMOS conversions are summarized in figure 1.3.5.2, and include the 8085, 6801, and the 6809 as well as the 16 bit 8086 and the single microcomputer chip, the 8048.

An example of the power-saving of CMOS over NMOS is shown in figure 1.3.5.3 which compares the National Semiconductor NSC800 (a CMOS version of the 8085 with the Z80 instruction set) with the 8085, 8080, and Z80. The NSC800 used in a system with memory and I/O uses about 15 times less power than the 8085. Note that the 8080, 1800, and the NSC800 have MIL-SPEC components.

FIGURE 1.3.5.1 8 BIT MICROPROCESSORS STILL USEFUL

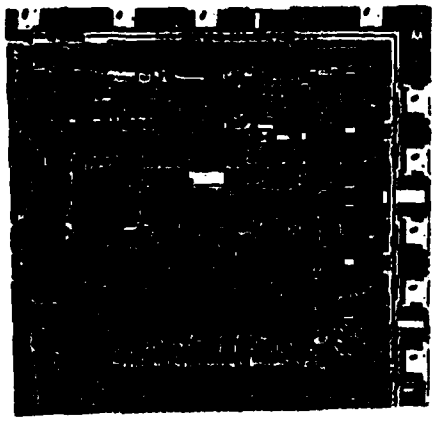
- THE MOST POPULAR FAMILIES INCLUDE
  - INTEL 8080/8085/8088
  - MOTOROLA 6800/6809
  - ROCKWELL 6500
  - ZILOG Z80

MOST HAVE MILITARY VERSIONS

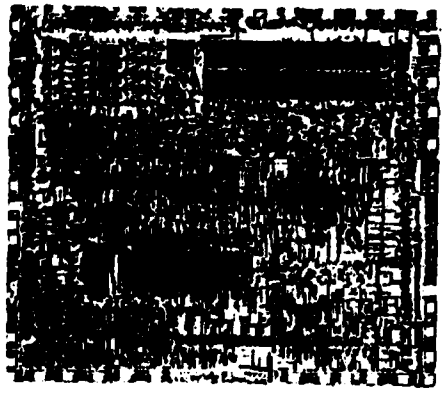
- CAPABILITIES SIGNIFICANTLY ENHANCED BY
  - AVAILABILITY OF MATURE SUPPORT SOFTWARE
  - SMART PERIPHERALS FOR I/O PROTOCOLS
  - DMA FOR MOVING LARGE BLOCKS OF MEMORY
  - PARALLEL PROCESSING ON MULTI-BUS
  - FLOATING POINT ARITHMETIC PROCESSOR CHIPS
- APPLICATIONS WIDE SPECTRUM IN GUIDANCE AND CONTROL
- LOW COST AND MATURITY WILL PREVENT REPLACEMENT
- VIRTUAL MEMORY MANAGEMENT EXTENDS ADDRESSABILITY RANGE

FIGURE 1.3.5.2 CMOS VERSIONS OF MICROPROCESSORS PROLIFERATING

MOTOROLA 146805E2 ROM-LESS MICROCOMPUTER



NATIONAL NSC 800
(Z-80 INSTRUCTION SET/8085 MUX MEMORY)



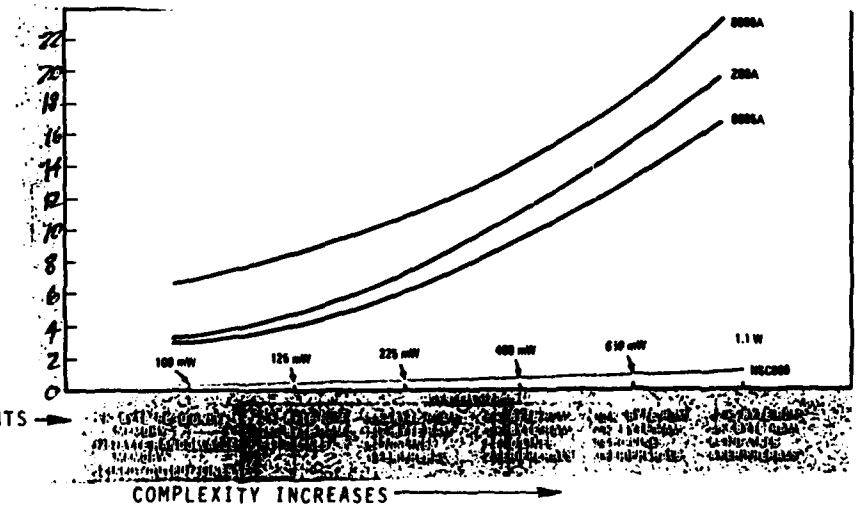
OTHER CMOS MICROPROCESSORS COMING

- NIPPON ELECTRIC 80C48
- HARRIS CMOS 8085 & 8086
- AMI CMOS 6809
- HITACHI CMOS 6801, 6805, AND 6805R2
- MOSTEK 38C70

FIGURE 1.3.5.3 P^2C-MOS ADDS NEW DIMENSION TO 8 BIT MICROCOMPUTERS LOW POWER

	8080	8085	Z80	6800	1800	NSC800
Technology	n-MOS	n-MOS	n-MOS	n-MOS	C-MOS	P^2CMOS ★
Power supply (V)	+12, +5, −5	+5	+5	+5	+3 to +12	+3 to +12
Cycle time (μs)	0.48/0.32	0.32/0.20	0.40/0.25	1.0/0.5	0.31/0.16	0.40/0.25
Instruction execution time (μs)	1.92/1.28	1.28/0.8	1.6/1.0	2.0/1.0	5/2.5	1.6/1.0
Bus structure (lines)	16 address, 8 data	8 address, 8 address or data	16 address, 8 data	16 address, 8 data	8 multiplexed address, 8 data	8 address, 8 address or data
Mil-spec components	yes	no	no	no	yes	yes
Memory refreshing	no	no	yes	no	no	yes

*EMULATES 8085
HAS Z-80
INSTRUCTION SET



1.3.6 Single Chip Microcomputers

The single chip microcomputer features are summarized in figure 1.3.6.1. The single chip microcomputers are used for dedicated controllers and processors and are also used as smart peripherals when programmed with appropriate firmware. The Intel 8048 has become a de facto standard for single chip microcomputers. This 11MHz clock, 8 bit microcomputer can perform a variety of controller functions. A CMOS version of the 8048 manufactured by National (the 80C48) is shown in figure 1.3.6.2. This unit operates at 6MHz and dissipates 25 mW during the operating mode and 100 microwatts during the idle mode. This unit will be available in November of 1981, and will cost $16 when purchased in large quantities. The Matushita MN 1500 4 bit single chip microcomputer boasts 8 bit performance, using an 8 bit data transfer in a single 2-microsecond cycle. The MN 1500 is shown in figure 1.3.6.3.

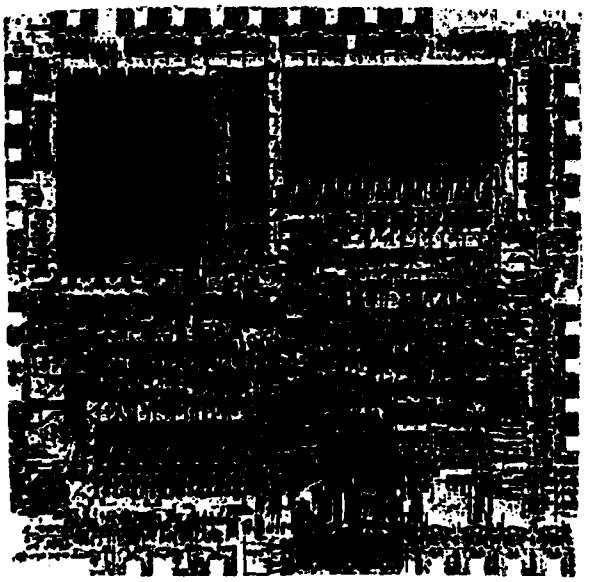
Another 8 bit single chip microcomputer is the Nippon MICRO PD 7801 illustrated in figure 1.3.6.4. The 7801 features a 128 Byte RAM, 4K Byte ROM, serial and parallel I/O wth a 2 microsecond instruction cycle.

The Motorola 6801 is a single chip microcomputer with 128 Byte RAM and 2K Byte ROM which can be expanded with off-board RAM, ROM, and I/O chips for systems requiring more capability than on the 6801. The 6801 also includes parallel and serial I/O.

FIGURE 1.3.6.1 IMPORTANCE OF SINGLE CHIP MICROCOMPUTERS

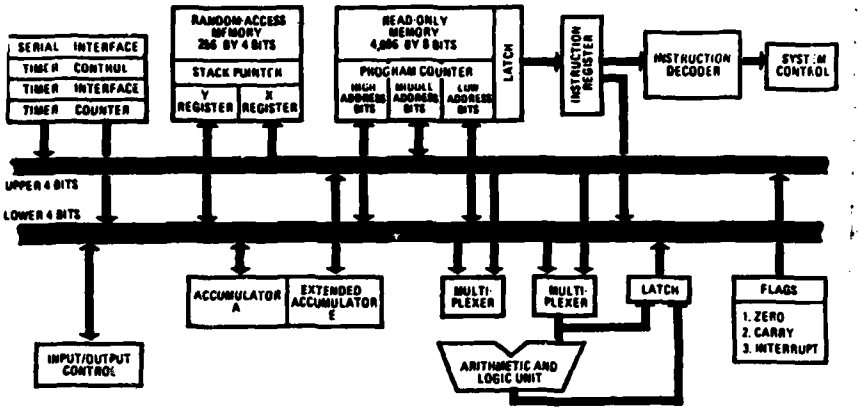
- USED FOR IMBEDDED CONTROLLERS / PROCESSORS
- SINGLE CHIP CONTAINS
  - CPU
  - RAM
  - EPROM or ROM
  - Digital I/O
  - Some Have Analog I/O
- SOME ARE EXPANDABLE TO ADD MEMORY / I/O
- BOTH 4 BIT AND 8 BIT VERSIONS
- EXAMPLE: 8 BIT 8048, 11 MHz CLOCK, 1.36 MICROSEC CYCLE TYPE HAS BECOME DEFACTO STANDARD

FIGURE 1.3.6.2 CMOS VERSION OF 8048 SINGLE CHIP MICROCOMPUTER



- National 80C x 48
- 6 MHz
- Low Power Idle Mode
  - Retains Data
- 5mA operating, 20 microamp Idle
- $16 each in quantity
- 1st delivery November 1981

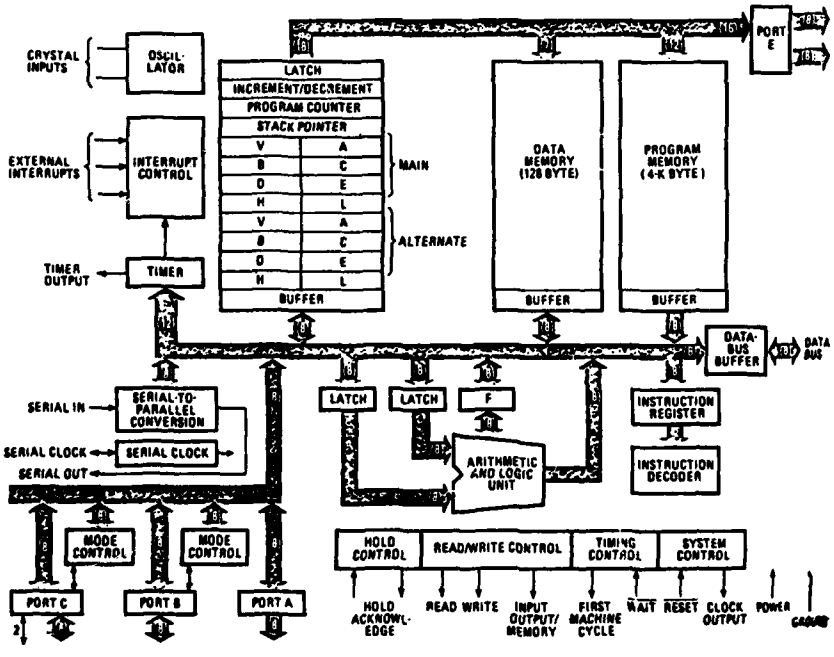
FIGURE 1.3.6.3
MATUSHITA MN 1500 4 BIT SINGLE CHIP MICROCOMPUTER
WITH 8 BIT PERFORMANCE



N-MOS

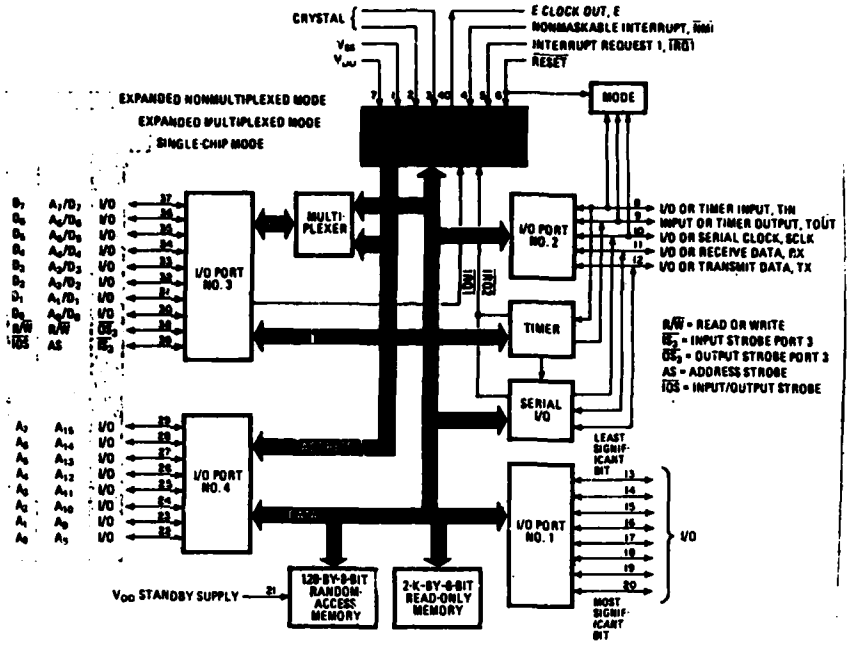
- 8 BIT DATA TRANSFER IN SINGLE 2 μSEC CYCLE
- 4 MICRON LITHOGRAPHY

FIGURE 1.3.6.4
NIPPON μPD 7801 8 BIT SINGLE CHIP MICROCOMPUTER



2 μSEC CYCLE TIME

FIGURE 1.3.6.5 MOTOROLA MC 6801 - 8 BIT SINGLE SHIP MICROCOMPUTER EXPANDABLE WITH OFF-BOARD ROM/RAM/I/O



1.4.0 MICROPROCESSOR MEMORIES

1.4.1 Random Access Memory (RAM)

A variety of memories are used in microprocessor-based systems which have a wide spectrum of access times and cost per bit of memory as illustrated in figure 1.4.1.1. The cost per bit of MOS RAMS are overly pessimistic showing 0.1 to 1 cent per bit. The cost of three supply voltage 16K RAMs is about .03 cents per bit, and 64K RAMs are about .08 cents per bit with projections to be .02 cents per bit by 1984.

FIGURE 1.4.1.1 ACCESS TIME AND COST PER BIT OF VARIOUS MEMORY TECHNOLOGIES

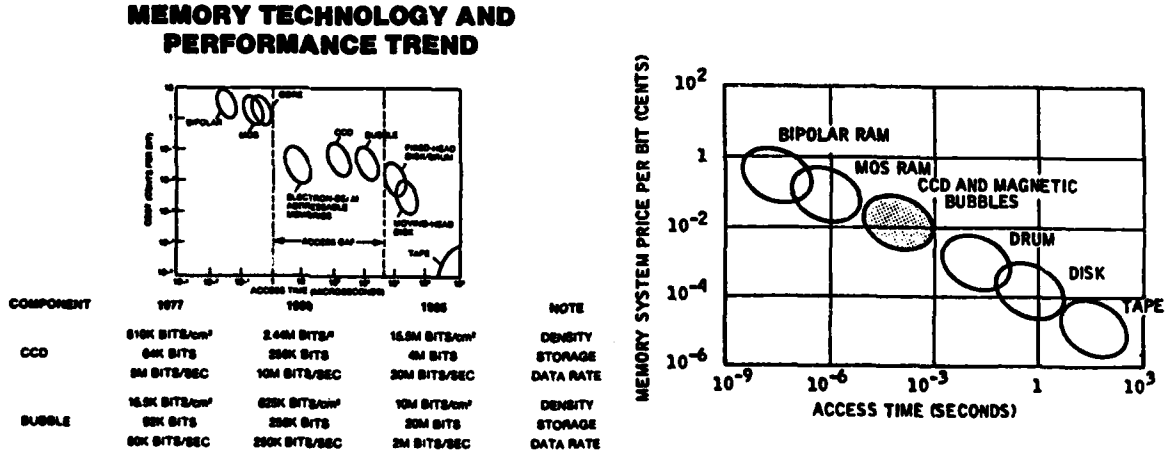


FIGURE 1.4.1.2 STATUS AND TRENDS IN RAMS

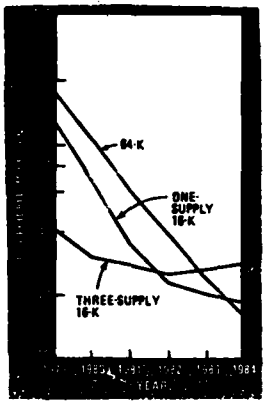
64K DYNAMIC RAMS TAKING OVER

Manufacturer	Part number	Die size[1] (mil²)	Access time[1] (ns)	Power dissipation[1] (mW)	Voltages (V)	Refresh (cycles/period)	B[2]	W[2]	32-K partial[3]	5-V 16-K[4]
Fujitsu, Tokyo, Japan	MB8164[5]	34,250	150/200	385/n.a.	+7[5], −2.5	128/2	FM	P	N	M
Texas Instruments, Dallas, Texas	TMS4164	33,000	150/200	125/17.5	+5	256/4	D	M	N	Y
IBM, Essex Junction, Vt.	internal	62,500	330–440	380/20	+8.5, +4.25, −2.2	256/2–3	D	M	Y[6]	N
Motorola, Phoenix, Ariz.	MCM6664[5]	39,000	150/200	275/30	+5	128/2	FM	P	Y	Y
Bells Labs, Murray Hill, N.J.	internal	61,800	170	440/n.a.	+8, −5	128/4	P	M	N	N
Hitachi, Tokyo, Japan	HM4864	46,460	150/200	330/20	+5	128/2	FM	P	M	Y
Mostek, Carrollton, Texas	MK4164	40,760	100/120	300/20	+5	128/2	P	M	Y[6]	Y
Mitsubishi, Tokyo, Japan	M58764S	41,750	150/200	250/27.5	+5	256/4	D	M	N	Y
National, Santa Clara, Calif.	NMC4164	31,000	120/150/200	200/20	+5	256/4	M	P	Y	Y
Toshiba, Tokyo, Japan	TMM4164C	38,800	120/150	250/20	+5	128/2	D	M	M	N
NEC, Tokyo, Japan	μPD4164	50,650	200	250/28	+5	128/2	D	M	N	Y
Intel, Santa Clara, Calif.	2164	n.a.	100/150/200	n.a.	+5	128/2	n.a.	n.a.	Y	Y
Siemens, Munich, West Germany	HYB4164	39,000	150	250/n.a.	+5	256/4	n.a.	M	n.a.	n.a.
ITT, Freiburg, West Germany	ITT4564	36,000	150	250/25	+5	128/2	D	M	Y	n.a.
AMD, Sunnyvale, Calif.	Am9064	< 40,000	100/150/200	200/20	+5	128/2	n.a.	n.a.	M	Y
Fairchild, Mountain View, Calif.	F64K	36,450	120	< 200	+5	256/4	M	P	n.a.	n.a.
Inmos, Colorado Springs, Colo.	n.a.	< 40,000	n.a.	n.a.	+5	256/4	n.a.	n.a.	N	Y
Signetics, Sunnyvale, Calif.	2164	40,700	60/80/120	300/15	+5	256/4	n.a.	n.a.	Y	n.a.

1. Actual and speculative values are intermixed. 2. B = bit-line material, W = word-line material, M = metal, D = diffused (ion-implanted), P = polysilicon, F = folded bit line. 3. Newer 8248 is 5-V only. 4. Not a partial. 5. Newer 8685 has no per-1 refresh. 6. 4332 contains 2 16-K RAMS, 4532 is a 64-K partial. 7. Y = yes, N = no, M = maybe.

$1B MARKET BY 1983

64K WILL BE CHEAPER/BIT BY 1983



256K & 1M RAMS ARE COMING

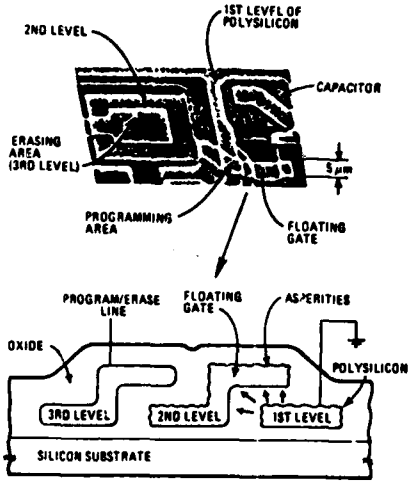
JAPANESE MAY BE FIRST

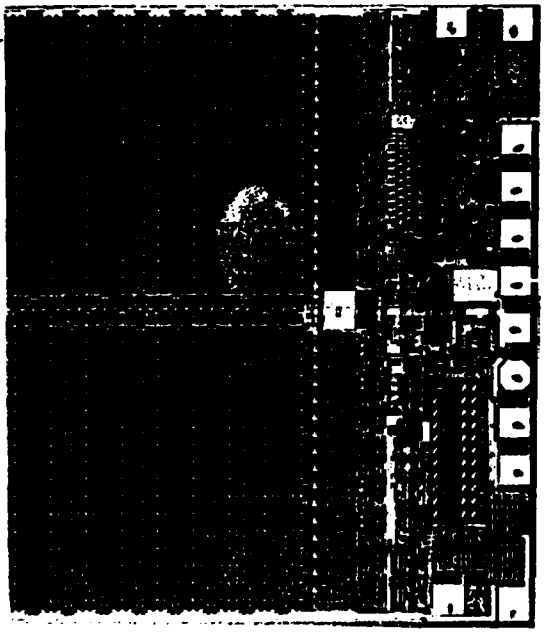
- o NIPPON ELECTRIC
- o MUSASHINO
- o KAWASAKI(NOW DEFUNCT)

FIGURE 1.4.1.3

- 8K STATIC MOS RAMS REPLACING HI SPEED BI-POLAR MEMORIES
  - MK 4118/MK 4801 USE POLYSILICON-LOAD TECHNOLOGY
- 1K NON VOLATILE RAM USES POLYSILICON
  - XICOR X2201/X2202

CELL EXPANDED





The 64K bit RAM came of age in 1981 with a number of types available which are summarized in figure 1.4.1.2. Most of the 64K RAMs use a single 5 volt power supply. The access time for the 64K RAMs varies from 100 to 200 nanoseconds with dissipation of 200 to 300 mW while operating and 20 to 30 mW in standby. The market for 64K bit RAMs is expected to be $1 billion by 1983.

Currently, the 16K RAM is cheaper per bit than the 64K RAM, but this trend is expected to reverse by 1983 when the 64K RAMs mature and become less costly on a per-bit basis.

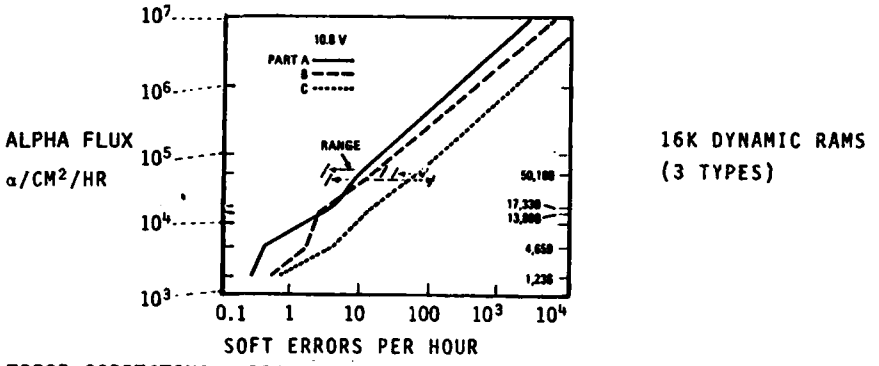
The next expectation in RAM memories is the introduction of the 256K RAM and the 1M bit RAM. The Japanese may be first with the new high-density RAM chips - perhaps Nippon Electric or Musashino.

Additional trends in RAMs are shown in figure 1.4.1.3 which shows new high speed 8K bit static MOS RAMs which are challenging high-speed bi-polar RAMs. These new static RAMs use the new polysilicon-based technology with the MK 4118 and the MK 4801 as examples. Xicor has introduced a new polysilicon non-volatile RAM with 1K bits, the X2201 and the X2202.

One problem with RAMs accentuated in the high density RAM has been soft errors due to alpha particles. Alpha particles are found in cosmic rays and natural radiation from traces of radioactive elements in the plastic cases of RAM chips. Figure 1.4.1.4 shows the error rates as a function of alpha particle flux for three types of 16K dynamic RAMs. Soft errors due to alpha particles can be eliminated by using error correcting codes such as the Hamming code. Soft errors can be reduced by using a thin dye coating on the RAM chips which cause transitions in the dye rather than errors in the MOS RAM.

FIGURE 1.4.1.4 ERROR CORRECTING RAM MEMORIES

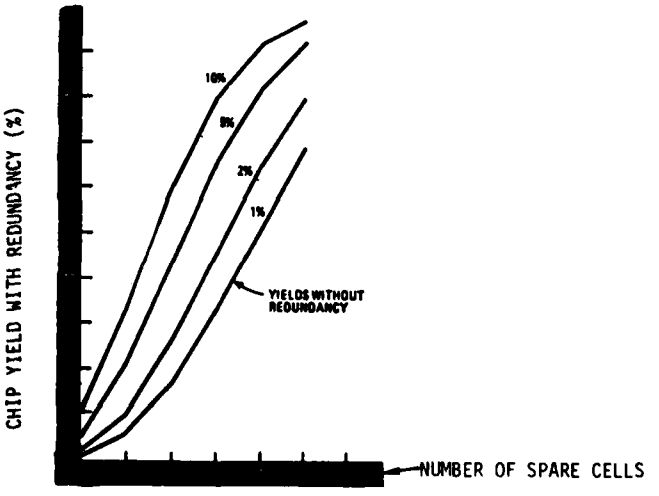
- ALPHA PARTICLES CAUSE SOFT ERRORS



- ERROR CORRECTING CODES SUCH AS HAMMING CODE

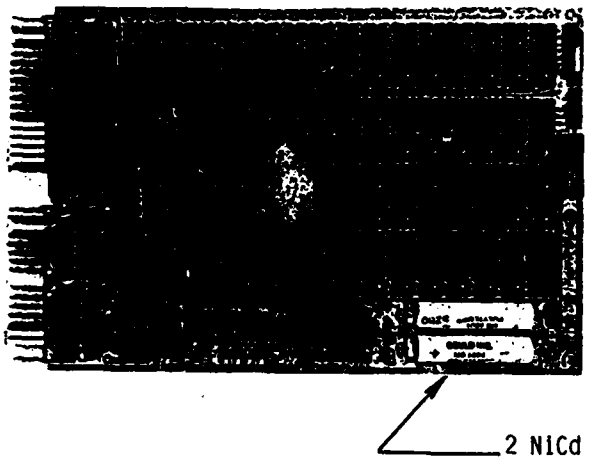
Figure 1.4.1.5 shows a technique used by RAM manufacturers to increase the yield of RAM chips by introducing a few redundant memory cells on the chip. For example, the use of 4 redundant memory cells increase the chip yield from 10% to 90%.

FIGURE 1.4.1.5 STATIC RAMS USE REDUNDANCY TO INCREASE YIELDS



MOS RAMs loose their memory when the power is turned-off, unlike the magnetic-core memories of the computers of the 1960's and the early 1970's. One technique for providing a pseudo non-volatile characteristic for MOS RAMs is the use of keep-alive batteries with the RAM maintained in a stand-by mode. An example of a 32K byte CMOS RAM circuit board uses two NiCd batteries (figure 1.4.1.6) to retain the memory for 64 hours after a power shutdown. This CMOS memory has an access time of 450 nanoseconds. Another technique for achieving a non-volatile RAM is the use of a non-volatile "shadow" (figure 1.4.1.7) using an MNOS (metal nitride oxide semiconductor) or polysilicon floating gate (FLOTOX) technology. The volatile MOS RAM transfers its memory contents to the non-volatile MNOS RAM upon detection of power shutdown. The shadow RAM retains the memory after power shut-down. Upon power start-up, the data is transferred to the MOS RAM and the MNOS RAM back-up (shadow) is erased to be ready for the next power-down. Figure 1.4.1.7 summarizes the characteristics of a number of non-volatile RAM memories which are slow to write but fast to read. The MNOS RAMs are sometimes referred to as EEPROM (electrically erasable PROM).

FIGURE 1.4.1.6 NON-VOLATILE RAM USING KEEP-ALIVE BATTERIES



- 32K BYTE C-MOS RAM
  - 64 Hour Retention After Power Down
- ADAC 1816 CMOS-16 for LSI-11
- 450 NS Access Time

2 NiCd Batteries

FIGURE 1.4.1.7 NON-VOLATILE RAM USING NON-VOLATILE "SHADOW"

PROCEDURE

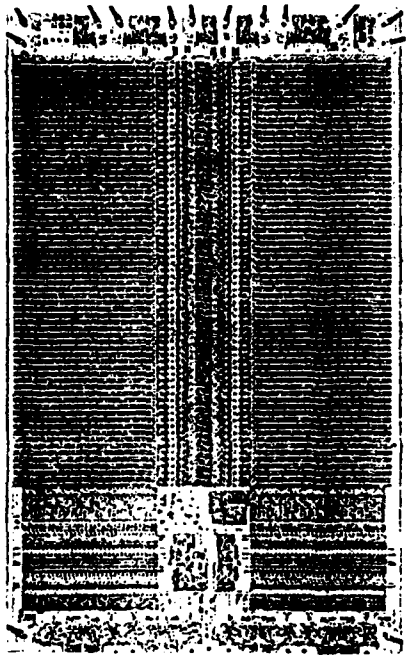
- RAM MEMORY TRANSFERS TO NON-VOLATILE ELEMENT (SHADOW) WHEN POWER FALTERS
- SHADOW RETAINS DATA WHEN POWER OFF
- ON START-UP DATA TRANSFERRED BACK TO RAM
- BACK-UP ERASED

	General Instrument Corp. ER1711	Hitachi Ltd.	SGS-ATES M 120	Toshiba Corp.	Xicor Inc. X2201, X2202
Organization	256-by-4-bit static	experimental dynamic	256-by-4-bit EE-PROM	16-bit experimental static	1-k-by-1-bit static
RAM technology	p-channel aluminum gate	n-channel polysilicon gate	complementary-MOS polysilicon gate	n-channel aluminum gate	n-channel polysilicon gate
Nonvolatile element	MNOS transistor	dynamic injection MNOS	polysilicon floating gate	MNOS capacitor	polysilicon floating gate
Elements/cell	8 transistors, 2 MNOS transistors	3 transistors	6 transistors	6 transistors, 2 capacitors	8 transistors, 1 capacitor
RAM voltages	+5 V, −12 V	+5 V	+5 V	+5 V	+5 V
RAM access time	900 ns	100 ns	450 ns	100 ns	250 ns
RAM-to-backup voltage Recall voltage Erase voltage	−21 to −17 V* −15 to −10 V* +25 to +30 V*	+25 V n.a. n.a.	+20 V n.a. n.a.	+30 V n.a. −30 V	+5 V +5 V +5 V
RAM-to-backup transfer time	1 ms/3 days of retention	50 ns/cell +1 ms	2 to 100 ms	1 ms	2 to 4 ms
Recall time	100 to 300 μs	0	0	0	1 μs
Endurance	10^4 cycles	–	10^6	10^5 cycles	10^3 – 10^5 cycles
Package	22-pin	n.a.	18-pin	n.a.	18-pin
Comments	*supplied by external dc-dc converter that needs only +5 and −12 V	small cell size (< 0.62 mil^2)	bit-alterable		X2201 has array recall X2202 has bit recall

1.4.2 Programmable Read-Only Memory (PROM)

A 16K CMOS PROM is illustrated in figure 1.4.2.1. This 16K PROM (Harris HA 6616) uses polysilicon fuse technology and has 125 to 250 nanosecond access time. The CMOS PROM operates over the military temperature range of -55 to +125 degrees C. It is projected that 64K CMOS PROMs will be available in 1982.

FIGURE 1.4.2.1 16K CMOS PROM



- 2K Bytes
- Harris HA 6616
- Polysilicon Fuse Technology
- 125-250 nanosec access
- OP PWR 25MW/MHz (400MW @ 8 MHz)
- STBY PWR 500 microwatt
- 2.5 micron lithography
- -55^{0} to $+125^{0}$C temperature range

64K CMOS PROMS in 1982

- About $65

The important trends in PROMs are shown in figure 1.4.2.2. EPROMs (ultra-violet erasable PROMs) are getting denser and faster with 64K EPROMS now available. The 2732 EPROM has an access time of 200 nanoseconds. Intel has introduced the electrically erasable 16K bit EEPROM (2816) which uses the FLOTOX process shown in figure 1.4.2.3. This direct replacement for the 2716 EPROM has the feature of individual byte erasability. The availability of 64K bit EEPROMS are projected for 1983.

FIGURE 1.4.2.2 STATUS AND TRENDS IN PROMS

● NEW FASTER DENSER CELL DESIGNS FOR 32K/64K E-PROMS

-INTEL
-TI
-MOTOROLA
-MOSTEK

● INTEL 2732A E-PROM

-H-MOSE
-32K
-3μ LITHOGRAPHY
-200NS ACCESS TIME

● INTEL 16K EE-PROM

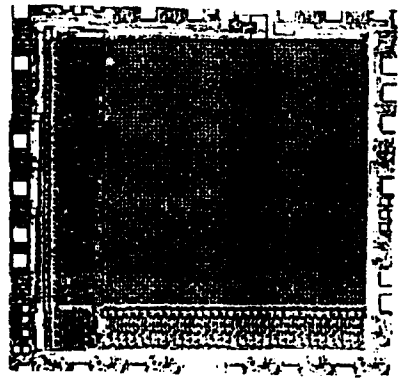
-REPLACEMENT FOR ULTRA-VIOLET ERASABLE 2716 E-PROM
-FLOATING GATE TUNNEL-OXIDE (FLOTOX) PROCESS
-ELECTRICALLY ERASABLE
-H-MOSE TECHNOLOGY

● INTEL 16K SUPER FAST PROM

-POLYSILICAN FUSE
-3μ LITHOGRAPHY
-25 NS ACCESS
-600 MW POWER
-BI-POLAR
-BURNED LINKS DON'T GROW BACK

● HITACHI 16K EE PROM

-HN 48016 (REPLACEMENT FOR 2716)
-350 NS ACCESS TIME

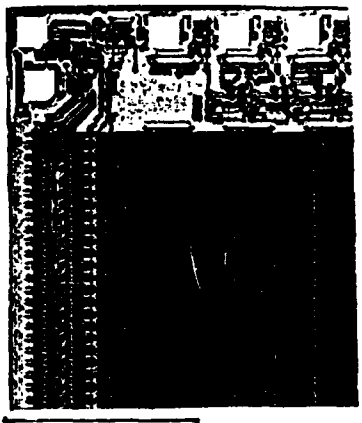


TECHNOLOGY VIABLE FOR 64K PROMS BY 1983

Other PROM developments are summarized in figure 1.4.2.3, which include the projection of 256K fused ROMs by 1982. The two types of EEPROMs are illustrated in figure 1.4.2.4. Two other EEPROM types are shown in figure 1.4.2.5, including the Hughes 8K CMOS EEPROM which is a direct replacement for the ultra-violet erasable 2708 EPROM.

FIGURE 1.4.2.3 IMPORTANT MEMORY DEVELOPMENTS

- 16K BITS ELECTRICALLY ERASABLE PROM'S (EEPROM)
  - INTEL 2816 - INDIVIDUAL BYTES ERASABLE



2K BYTES

- HITACHI 48016 250NS ACCESS/USES POLYSILON GATES INSTEAD OF METAL GATES

IMPROVEMENT OVER MNOS TECHNOLOGY

RETAINS DATA OVER 10 YEARS

IN 1982 EXPECT ➡ ⇨ 256K ROM'S, 64K UV EPROMS, 256K RAM'S (?)

- PROTECTIVE DIE COATING PROTECTS RAM'S FROM ALPHA PARTICLES
  - 64K SOFT ERROR RATES OF 1000 FAILURES/10^9 HOURS (FITS)
  - WITHOUT DIE COATING SOFT ERROR RATES 10-100K FITS

FIGURE 1.4.2.4 N-CHANNEL MNOS INCREASE SPEED/DENSITY OF EEPROM'S

16K BITS EEPROM'S

	MNOS – silicon-gate	Floating gate
Power supplies for read mode and write/erase mode	+5 V, 25 V	+5 V, 20–25 V
Write time	1 ms/word	10 ms/word
Erase time	100 ms	10 ms
Write/erase cycles	10^4	10^4
Read cycles	unlimited	unlimited
Data retention	10 years	10 years
Power dissipation in read mode and write/erase mode	210 mW, 315 mW	350 mW, 500 mW
Access time	140 ns typical	200 ns typical
Cell size	0.52 mil^2	0.85 mil^2
Chip size	177 by 157 mil^2	165 mil^2

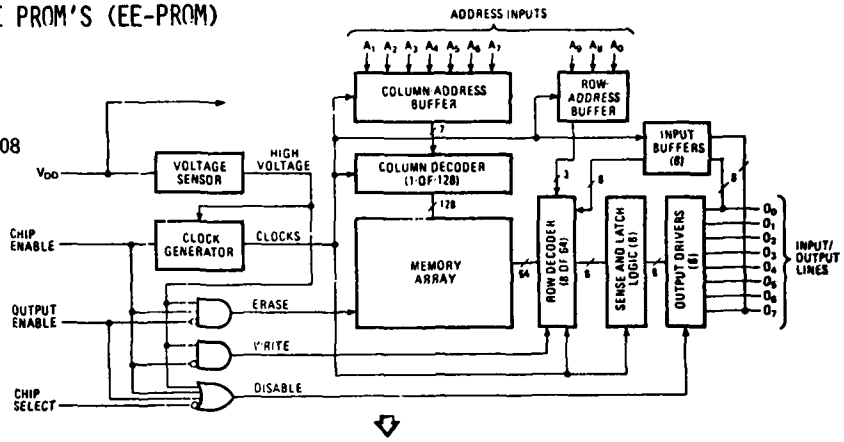
N-Channel Metal Nitride Oxide Semiconductor
MNOS
General Instruments

N-Channel Floating Gate
Polysilicon Oxide
FLOTOX
INTEL

FIGURE 1.4.2.5

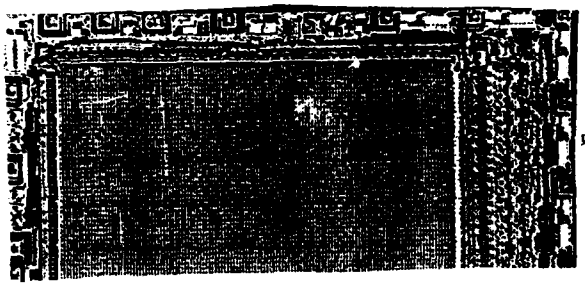
- ELECTRICALLY ERASABLE PROM'S (EE-PROM)
- 8K EE-PROM

-HUGHES 8K HNOV0463
-CMOS PROM
-DIRECT REPLACEMENT FOR 2708
-10mw READ/500μw STDBY
-READ 600ns
-WRITE 100μs
-REPROGRAM 100ms
-$400



- 16K EE-PROM

-HITACHI HN-48016
-MNOS
-REPLACEMENT FOR 2716
-350ns READ
-5-25volt ERASES



½HNOV0463

A new fast NMOS EAROM (electrically alterable ROM) is shown in figure 1.4.2.6. This 1K EAROM can be reprogrammed in 300 nanoseconds compared with 22 milliseconds for the older type of NMOS non-volatile RAM. This non-volatile RAM is estimated to hold its memory for 100 years, compared with 10 years for older MNOS RAMs.

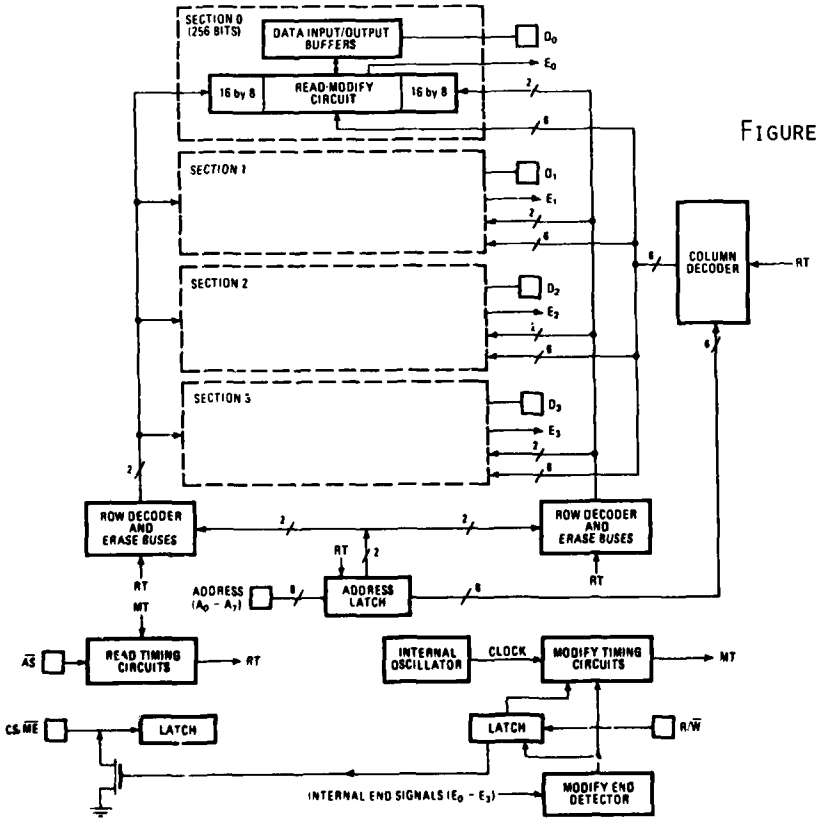


FIGURE 1.4.2.6 FAST MNOS EAROMS

SGS-ATES (ITALY) M120

1K BIT
4K BIT COMING

- CHANGE IN 300 NS

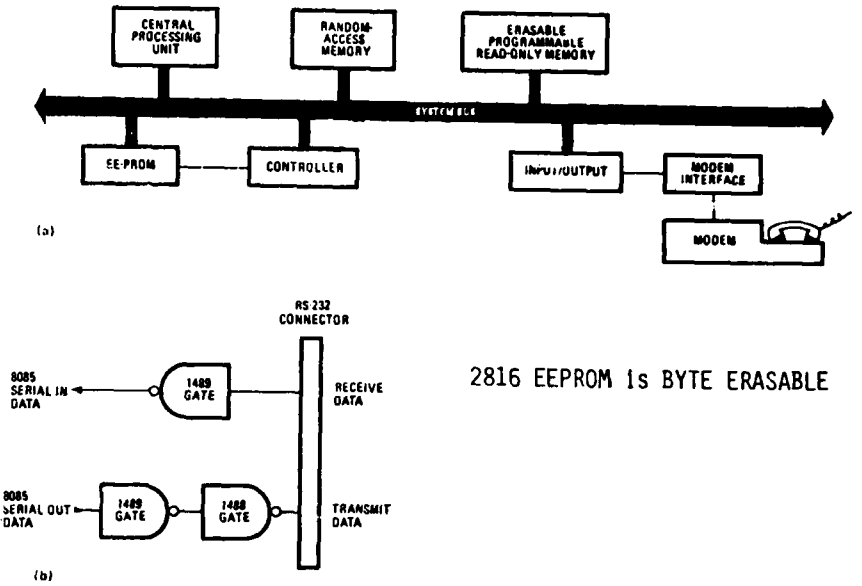
(OLD TYPE MNOS REQUIRED 22MS TO CHANGE)

FAST NON VOLATILE RAM

(HOLDS CHARGE FOR 100YRS)

The 2816 EEPROM which is byte erasable can be employed in systems requiring firmware to be changed remotely (figure 1.4.2.7).

FIGURE 1.4.2.7 EEPROM'S PERMIT FIRMWARE TO BE CHANGED REMOTELY



2816 EEPROM is BYTE ERASABLE

1.4.3 Magnetic Domain Bubble Memories

Bubble memories have long held the promise of high density, low cost per bit, non-volatile memory with slow access as a replacement for mechanically rotating mass memory (e.g. disks). However, bubbles have been a disappointment up to the present, even though several semi-conductor companies have successfully achieved 1M bit bubble memory chips. These bubble memories remain expensive with a 128K byte bubble memory card costing in the range of 3 to 5000 dollars.

The disappointment in bubble memories has been compounded by three of the leading competitors (i.e. Rockwell, TI, and National) dropping out of the race during the last six months of 1981. Bubble memory chips are still commercially available from Intel, Hitachi, and Plessey. IBM is actively developing a new type of bubble called the contiguous-element disk which has the potential for greater density than the traditional chevron-type bubble memory.

The concept for a bubble memory is shown in figure 1.4.3.1. This figure illustrates the Rockwell chip which uses the chevron configuration. The bubble loops contain a string of magnetic bubbles which circulate along the loops and are caused to rotate by a DC magnetic field orthogonal to the plane of the bubble loops. The bubbles are generated by the generator loop on the left - a bubble represents a zero. As a bubble passes the detector loop, it generates a voltage due to the bubble magnetic flux cutting the detector loop ($e = d\phi/dt$). The annihilator loop is used to erase a bubble. The characteristics of a 1.6M bit space qualified bubble memory delivered to NASA Langley in 1978 is shown in the figure. This memory used 102K bit bubble chips. Subsequently, Rockwell, Intel, and others developed a 1M bit bubble chip.

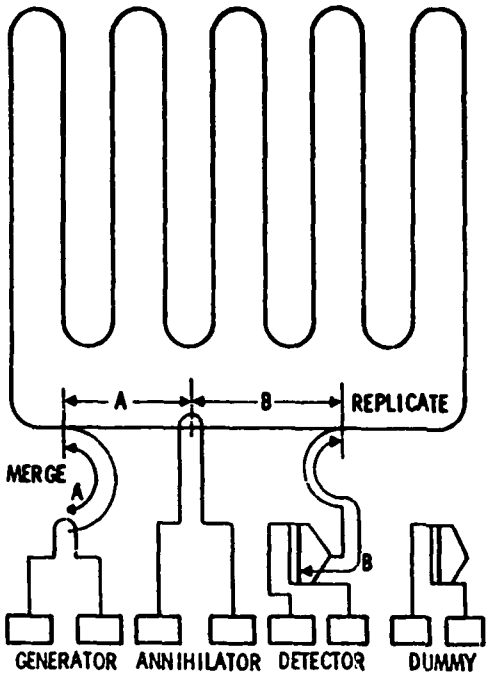
The status of bubble chips in August 1979 is shown in figure 1.4.3.2. Intel was the first to achieve the 1M bit bubble and continues as the leader in this technology with Rockwell/TI/National dropping out of the commercial competition.

The new contiguous-element bubble "disk" is illustrated in figure 1.4.3.3 and compared with the traditional permalog chevron bubble structure. The contiguous element bubble structure is being developed by IBM with a 4M bit chip currently under development with a 16M bit chip planned by 1986. The storage density of the contiguous element chip is 9.6 times that of the chevron element chip.

Other bubble memory developments are shown in figure 1.4.3.4 with a Plessey bubble memory board with 512K bytes for the SBC-80 chassis selling for $3010. Figure 1.4.3.5 shows two other bubble memory boards currently available from TI and Intel (768K bytes and 128K bytes, respectively).

Finally, some projections for bubble memories are presented in figure 1.4.3.6 which shows the ultimate limit for a bubble memory chip may be 256M bit chips. If this density is achieved, a memory board the size of the Intel SBC-30 would hold 32M bytes of memory. If and when this memory density is achieved, bubble memories may yield the promise of low-cost, mass memories for microprocessors.

Figure 1.4.3.1 BUBBLE DOMAIN MEMORY SYSTEM



ROCKWELL MEMORY DELIVERED TO NASA LANGLEY, MAY 1978

16 102K BUBBLE CHIPS (0.25 X 0.25 IN.) WITH 1.6M BIT MEMORY

ROCKWELL CHIP ARCHITECTURE

- REPLICATE FUNCTION CREATES DUPLICATE STREAM OF BITS
- PERMITS READING DATA WITHOUT REMOVING FROM STORAGE LOOP
- "A" PATHS & "B" PATHS EQUAL FOR PROPER BIT SPACING
- DUMMY HELPS CANCEL MAGNETORESTRICTIVE NOISE
- COMPLETE SYSTEM:
  - 12.75" X 12.7" X 5.3"
  - 47 LB.

FIGURE 1.4.3.2 RECENT DEVELOPMENTS

BUBBLE MEMORY CHIPS

COMPANY	CHIP CAPACITY	ORGANIZATION	DETECTION SCHEME
TI	256 K	256 K x 1 BLOCK REPLICATE/SWITCH	CONSECUTIVE BIT
ROCKWELL	256 K	256 x 1 BLOCK REPLICATE/SWITCH	ALTERNATE BIT DUAL DETECTOR
FUJITSU	256 K	256 x 1 MAJOR-MINOR/TRANSFER	CONSECUTIVE BIT
INTEL	1 M	256 x 4 MULTI BLOCK REPLICATE/SWITCH	CONSECUTIVE BIT

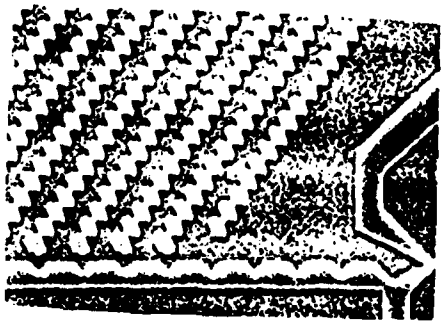
CONVENTIONAL CONFIG

- ASYMMETRICAL CHEVRON
- UNI DIRECTION PROPAGATION
- LITHOGRAPHY 2/3 BUBBLE DIAM

NEXT GENERATION

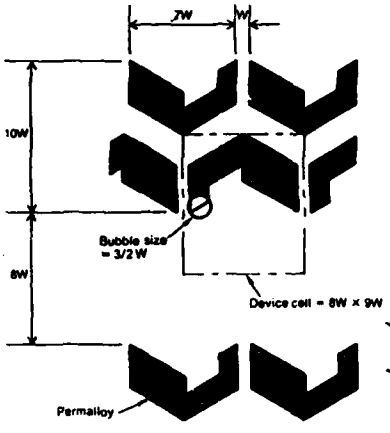
- CONTIGUOUS DISK TECHNOLOGY
- ION IMPLANTED PATERNS
- BI DIRECTIONAL PROPAGATION
- LITHOGRAPHIC RESOLUTION RELAXED

FIGURE 1.4.3.3 CONTIGUOUS - ELEMENT BUBBLE MEMORIES INCREASE STORAGE 10 FOLD



IBM 4MBIT CHIP UNDER DEVELOPMEMT
16MBIT BY 1986

EXISTING PERMALOY CHEVRON

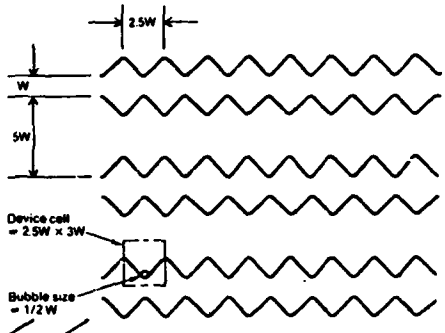


STORAGE DENSITY RATIO

$$\frac{\text{EXISTING}}{\text{NEW}} = \frac{72W^2}{7.5W^2}$$

$$= 9.6 \text{ times}$$

NEW CONTIGUOUS-ELEMENT DISK

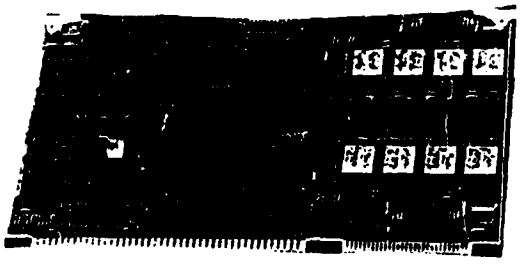


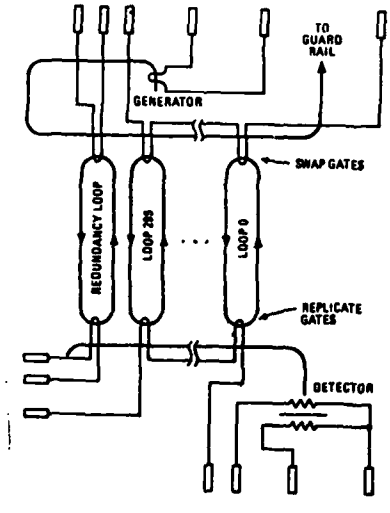
1Mbit/cm² — 1μ BUBBLE

4Mbit/cm² — 2μ BUBBLE

FIGURE 1.4.3.4 BUBBLE MEMORIES

- PLESSY BOARD FITS INTEL SBC-80 CHASSIS
  - 64K BOARD PBM 805
  - 512K BOARD, $3010

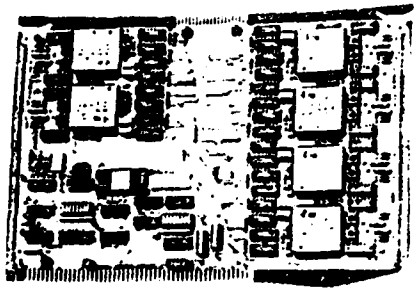




- TIB 1000 1M BIT BUBBLE MEMORY CHIP

FIGURE 1.4.3.5 BUBBLE MEMORIES

- TI TM 990/211 BOARD
  768 K BYTES, $15,200
  - -DATA TRANSFER RATE 85K BPS
  - -AVG ACCESS 11.2 ms
  - -6 1M BIT CHIPS
  - -8-10 WK DELIV
- INTEL iSBC 250 BOARD
  - -128 K BYTE
  - -AVG ACCESS 40 ms
  - -$4700



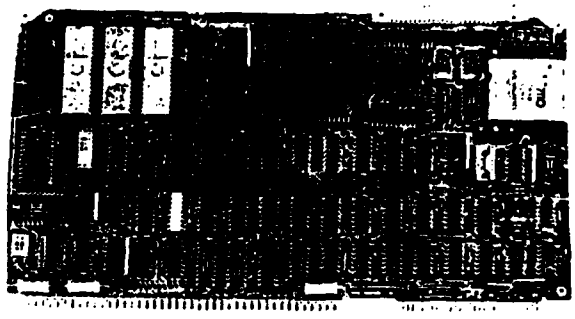


FIGURE 1.4.3.6
BUBBLE MEMORIES TO BECOME DENSER/FASTER

CURRENT	NEXT	ULTIMATE
1 MB CHIP	4 TO 16 MB CHIPS	256 MB CHIPS
	10 MM x 10 MM CHIP	

METHODS TO INCREASE DENSITY/SPEED

- SCALE DOWN CURRENT CONFIGURATIONS
- WALL ENCODED LATTICES
- CONTIGUOUS DISK MEMORIES
- CURRENT-SHEET DEVICES (10-20 TIMES FASTER CLOCK)

1.5.0 MICROPROCESSOR PERIPHERAL DEVICES

The availability of smart peripheral chips reduces the busy work of microprocessor CPU's and enhances throughput through the inherent parallel processing provided. Some of the smart peripherals available are listed in figure 1.5.0.1. One of the recent entries for smart peripherals is the memory management chips which expand the addressability of the CPU using virtual memory techniques.

FIGURE 1.5.0.1 SMART PERIPHERAL CHIPS

- REDUCE "BUSY-WORK" LOAD OF PROCESSORS
  - Enhances System Through-put
- GENERAL PERIPHERALS INCLUDE:
  - Floating Point Math Processors
  - Bus Interface Protocol / Controllers
  - Specialized I/O Controllers
  - Encryption Algorithms
  - CRT / Keyboard Controllers
  - Memory Controllers

Generally Based on Single Chip Microcomputer

ROM Program "Customizes" Peripheral Chip

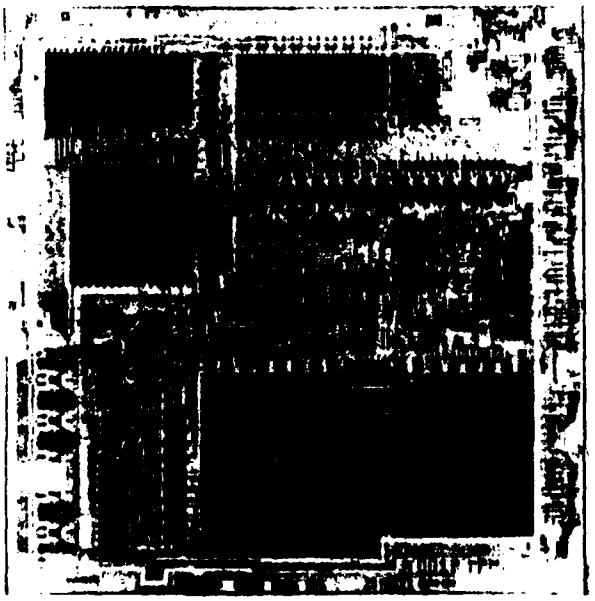
1.5.1 Arithmetic Processor Chips

A summary of some of the available arithmetic processor chips which provide parallel floating point arithmetic processing is given in figure 1.5.1.1. The AMD 9511 is the first of these floating point arithmetic processing units (APU) which provides 32 bit floating point. Its repertoire of functions includes the transcendental functions as well as the four basic arithmetic operations.

The AM 9512 is a 64 bit floating point chip oriented to data processing and does not include the transcendentals.

The 8087 is also a 64 bit floating point processor intended for use as a co-processor with the Intel 8086 16 bit microprocessor.

FIGURE 1.5.1.1 ARITHMETIC PROCESSING CHIPS



AM 9511 FIRST NAVIGATION, FOURIER TRANSFORMS, REAL TIME SYSTEMS. 32 BIT FLOATING POINT

AM 9512
64 BIT FLOATING POINT
SLOWER, DATA PROCESSING ORIENTED.

INTEL 8087

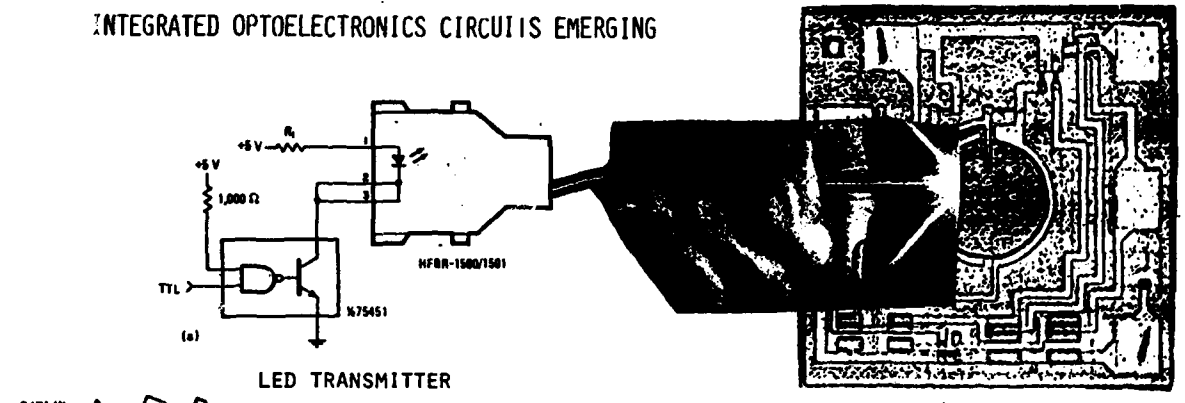
HIGH SPEED
FLOATING PONT

1.5.2 Bus Interface Controllers

A number of Bus interface controller chips are available which implement the protocol of such busses as the MIL-STD-1553B 1M bit/second serial data bus, the IEEE-488 16 bit parallel bus, the ARINC 424, the RS-232C serial interface, and others. Another trend is the use of fibre optics data busses for high bandwidth busses. Integrated optoelectronics is a new interface technology which provides the optical/electrical and electrical/optical interface for fibre optics busses as shown in figure 1.5.2.1.

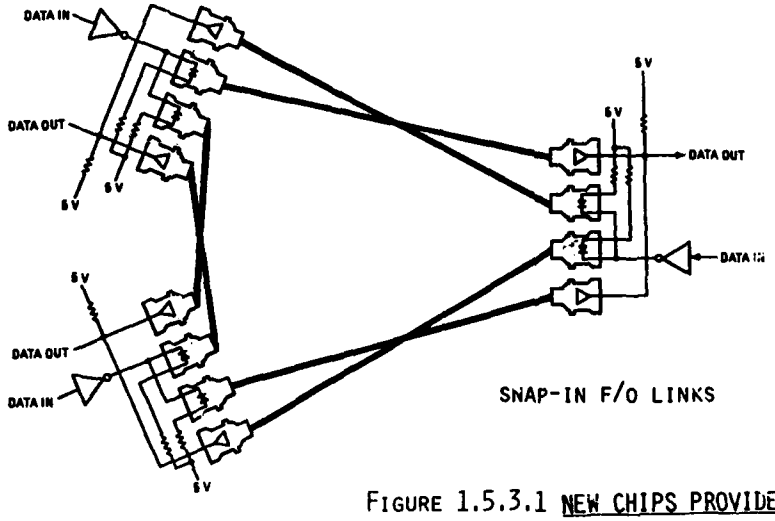
FIGURE 1.5.2.1 FIBRE OPTICS DATA BUS AVAILABLE

INTEGRATED OPTOELECTRONICS CIRCUITS EMERGING



LED TRANSMITTER

MFOD403F RECEIVER

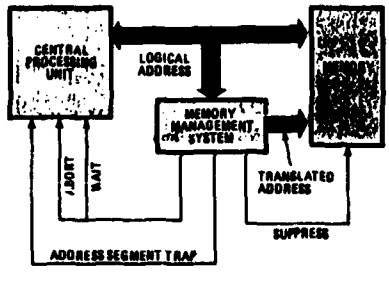


SNAP-IN F/O LINKS

- STANDARD SERIAL DATA BUS
  - RS232C
  - ARINC 424
  - MIL-STD-1553
  - RS422 OR 429 CAN BE IMPLEMENTED

FIGURE 1.5.3.1 NEW CHIPS PROVIDE VIRTUAL MEMORY MANAGEMENT

Z-8003 Companion to Z-8000



Memory Management Terminology

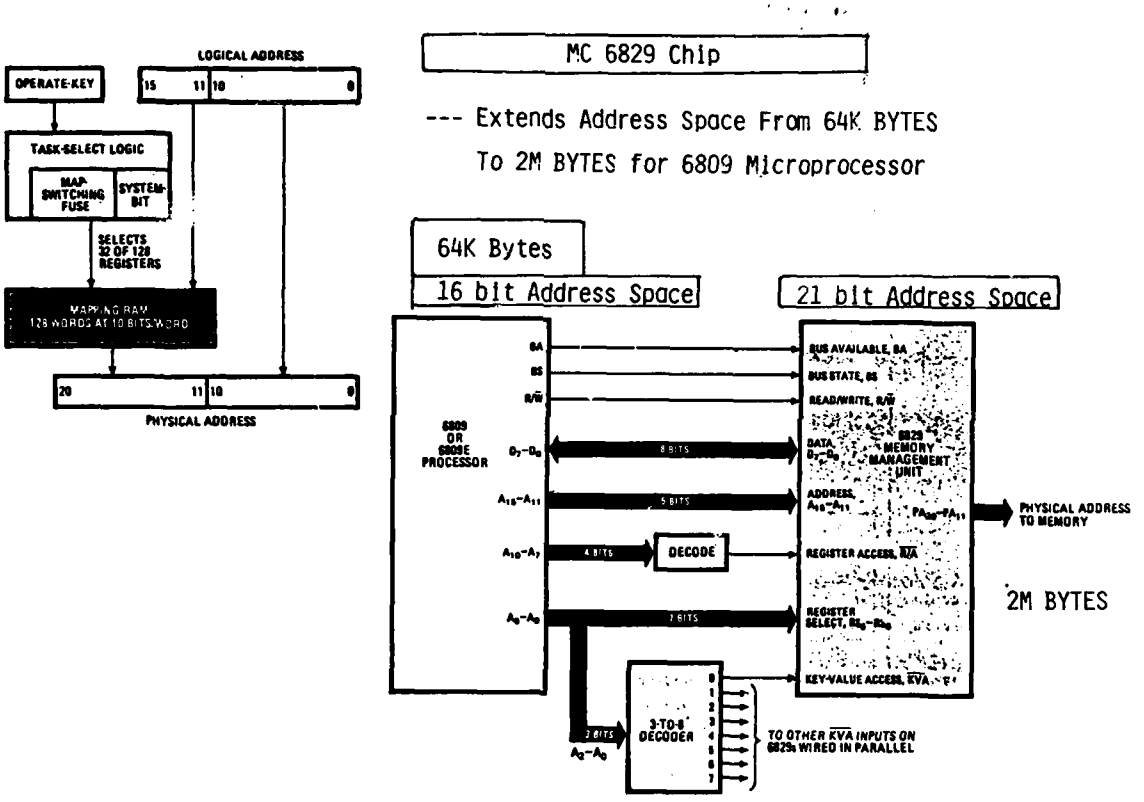
abort (instruction)	halting an instruction because its operands are not in main memory
address binding address mapping address translation	calculating the real physical memory address of operands from the logical address supplied by the processor
bank switching	wiring memory banks in parallel so that only one is active at any given time (under software control)
base address	the address of the first location in a segment or page of memory
content-addressable memory (CAM)	a memory bank in which the contents of each location are simultaneously compared with incoming data for matches
compaction	relocating programs in memory so that they and unused areas are contiguous
descriptor	a preformatted data set describing the location, access privileges, and status of a region of memory
dirty regions	regions of memory that have been modified since being loaded into main memory
dynamic allocation	the run-time reorganization of data storage to accommodate an executing process
hit ratio	the percentage of time that the information requested is resident in a cache memory
latency	the extra time delay introduced by a memory management unit into the path that the address lines take to main memory
logical address	the address used in programs to seperate logically distinct values
offset	the number of locations from the base address to the desired information
page	fixed-size region of memory
physical address	the bit pattern applied to the address bus of real physical main memory
residency	the situation when information currently needed is already in main memory
static allocation	the load-time organization of data storage that then stands all the while a program is executing
swapping	exchanging information between mass storage and main memory

1.5.3 Memory Management Controllers

An important new type of peripheral chip is the virtual memory management chip which permits a microprocessor to address a larger memory space than that permitted by the address bus width. Figure 1.5.3.1 is an example memory management chip for the Z8000 (i.e. the Z8003). The terminology for memory management is summarized in the figure.

The 8 bit microprocessors are generally limited to 64K bytes of memory space. The new memory management chips expands this memory space considerably. For example, the 6829 memory management chip shown in figure 1.5.3.2 expands the memory space to 2M bytes for the 6809 8 bit microprocessor.

FIGURE 1.5.3.2 MEMORY MANAGEMENT CHIP FOR 8-bit MICROPROCESSORS



1.5.4 Smart Peripheral Chips

Examples of smart peripheral chips are shown in figure 1.5.4.1 which join the early peripheral chips such as the CRT/keyboard controller chips, which were introduced almost concurrently with the early microprocessors. The Z8 is a single chip universal peripheral controller which can be programmed for a variety of dedicated controller functions which supports both 8 bit and 16 bit systems.

Another peripheral chip is the Z8065 chip which detects and corrects 12 bit burst errors during mass memory read cycles. Another type of memory error correcting chip is the AMD 2960, which detects and corrects RAM errors using a modified Hamming code. This chip uses ECL technology and is TTL compatible. Additional smart peripheral chips are illustrated in figure 1.5.4.2, including communication protocol chips such as the Fairchild F6856, and the AMI S6854 which supports ADCCP, HDLC, and SDLC data link protocols.

The AM S2811 is a super high speed arithmetic processor chip compatible with the 6800 microprocessor which multiplies two 16 bit numbers in 300 nanoseconds.

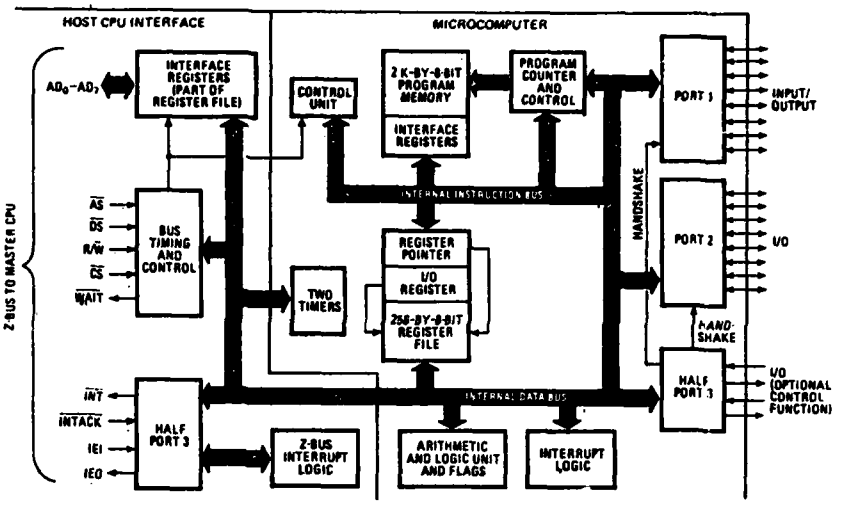
Figure 1.5.4.2 shows a four chip programmable peripheral controller using I^2L technology. This chip may be programmed to interface with a variety of mass memories, line printers, and CRT's.

FIGURE 1.5.4.1 SMART PERIPHERAL CHIPS

- Z8 SINGLE CHIP MICROCOMPUTER, Z-UPC

UNIVERSAL PERIPHERAL CONTROLLER

SUPPORTS 8 BIT AND 16 BIT SYSTEMS



- ERROR CORRECTING CHIPS

-Am Z8065 N-MOS SINGLE CHIP DETECTS/CORRECTS 12 BIT BURST ERRORS

- BURST ERRORS OCCUR DURING MASS MEMORY READ CYCLE

-ADVANCED MICRO DEVICES, INC. AMD 2960 DECTECTS/CORRECTS RAM ERRORS

- USES MODIFIED HAMMING CODE
- HANDLES 16 BIT LONG DATA FIELDS
- ECL TECHNOLOGY/TTL COMPATIBLE

FIGURE 1.5.4.2 SMART PERIPHERAL CHIPS

- COMMUNICATION PROTOCOLS

-AMI S6854 DATA LINK CONTROLLER ADCCP/HDLC/SDLC PROTOCOLS
-FAIRCHILD F6856 COMMUNICATION PROTOCOL CHIPS

- AM S2811 HIGH SPEED MATH, COMPATIBLE WITH 6800

-ADDS TWO 16 BIT NUMBERS IN 40 NS
-MULTIPLIES TWO 16 BIT NUMBERS IN 300 NS

- FOUR CHIP PROGRAMMABLE PERIPHERAL CONTROLLER, I^2L TECHNOLOGY

-FLOPPY DISKS

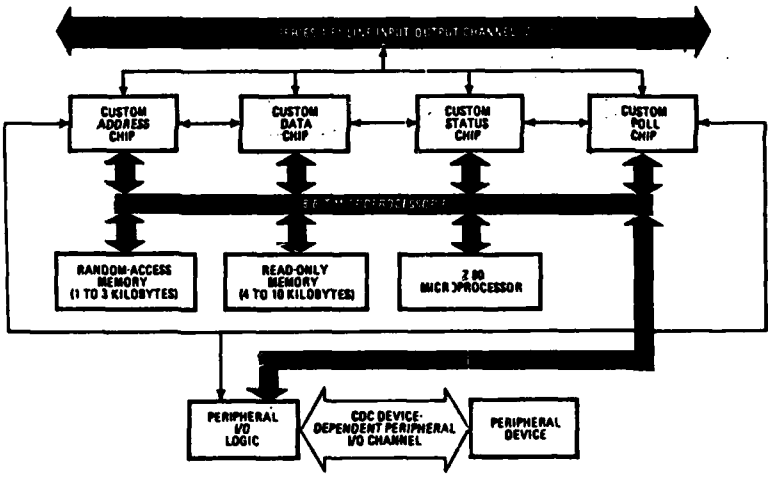
-CRT's

-HARD DISKS

-LINE PRINTERS

-USES Z80 MICROPROCESSOR

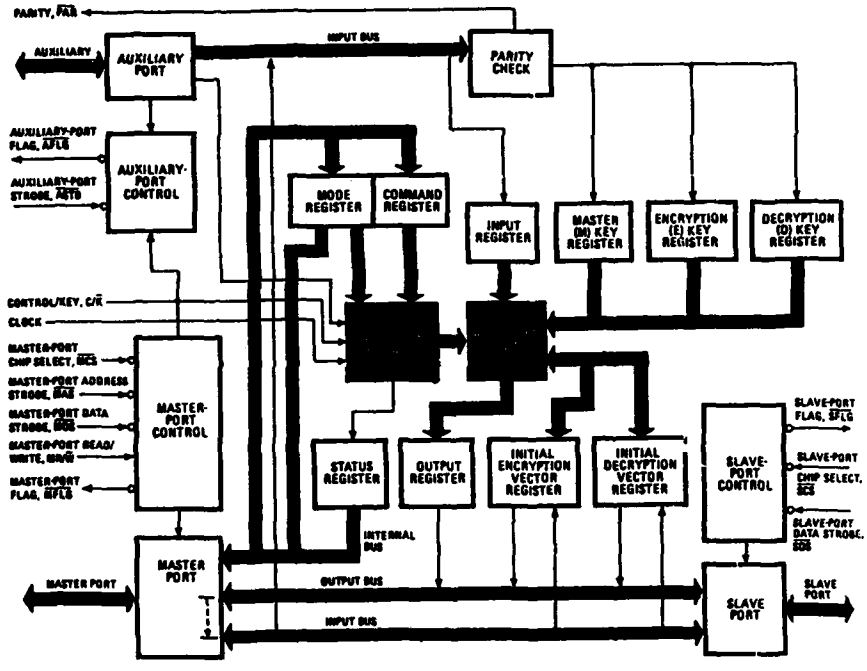
-ROM CONTROLS PERIPHERAL FUNCTIONS



1.5.5 Data Encryption Chips

In many applications, special security precautions are needed for the data transmitted between microprocessors. The NBS has defined a data encryption standard (DES) patented by IBM, but licensable to any user with no license fee (e.g. MILCO has received a license from IBM). A recent announcement (summer of 1981) is that the AM 9518 (figure 1.5.5.1) encrypts data on the fly to mass memory at a rate of 14Mb per second. The 9518 implements the NBS DES and is controlled by a standard microprocessor.

FIGURE 1.5.5.1 SINGLE CHIP DATA ENCRYPTION



- AM 9518
- 14 Mb/sec Encryption Rate
- Implements NBS DES (Digital Encryption Standard)
- Controlled by Standard Microprocessor
- Can encrypt/decrypt data on Fly to Mass Memory

Two other NBS DES chips are shown in figures 1.5.5.2 and 1.5.5.3, which depict the TMS 9940 capable of 4800 bit/second and the Intel 8294 capable of 640 bit/second encryption rate.

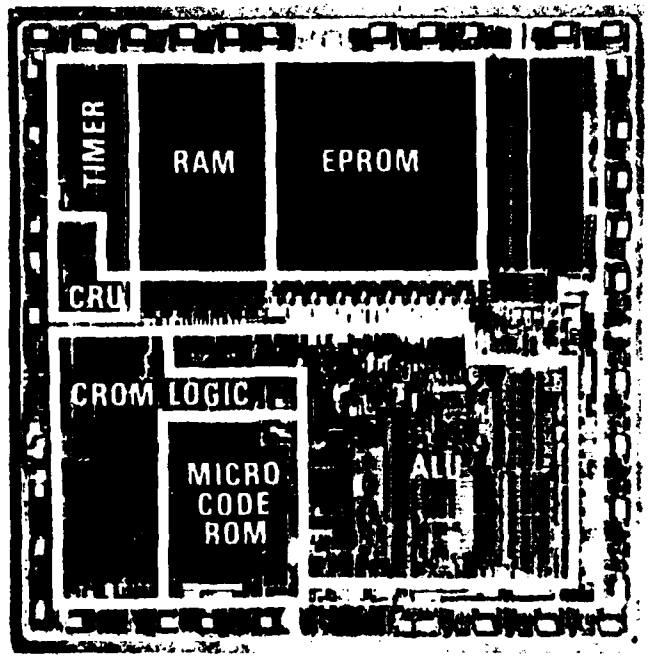
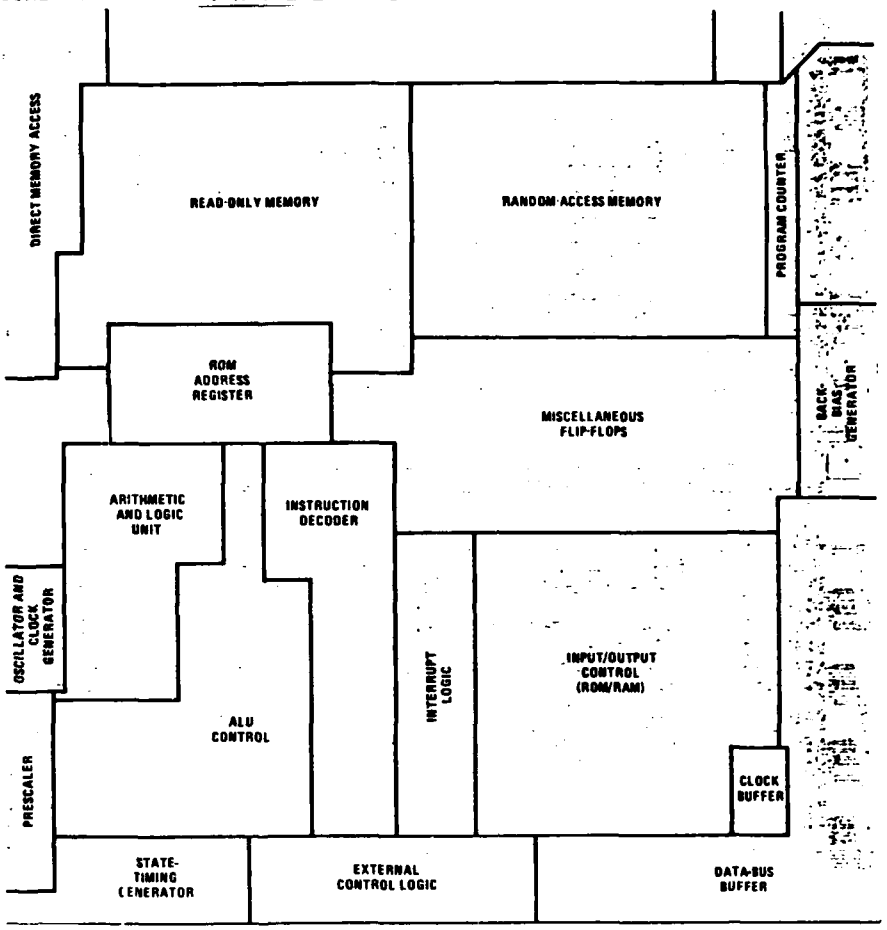


FIGURE 1.5.5.2
TMS 9940 DATA ENCRYPTION CHIP

NBS STD

4800 BIT/SEC

FIGURE 1.5.5.3 INTEL 8294 DATA ENCRYPTION CHIP



1.5.6 Analog Interfaces

The analog world requires analog to digital converters (ADCs) and digital to analog converters (DACs) with microprocessors. Some typical ADCs and DACs are shown in figure 1.5.6.1. As may be seen, ADC/DAC chips are available at a modest cost.

Figure 1.5.6.2 depicts a high speed ECL DAC which converts in 10 nanoseconds and a high resolution 18 bit DAC which converts in 2 microseconds. The DAC-320-18 is implemented using hybrid circuit technology and is available in military temperature range requirements.

A 16 bit resolution ADC is shown is figure 1.5.6.3, providing a conversion in 35 microseconds. A very high speed 12 bit DAC which sells for $108 is shown in figure 1.5.6.4. This DAC from Burr-Brown settles in 35 nanoseconds.

A superfast 6 bit resolution ADC which converts in 500 picoseconds (i.e. 0.5 nanoseconds) is shown in figure 1.5.6.5 which uses JJ technology. The figure also depicts a 14 bit resolution DAC.

An interesting new 8 bit analog microcomputer is the Intel 2920, having analog input and analog output, but uses a high speed pipeline processor to implement digital filters (see figures 1.5.6.6 and 1.5.6.7). The 2920 is capable of programming a shaping filter with 10 poles and 10 zeros in the S-plane. The speed is such that the sampling delays can be neglected (i.e. one can assume the filter is in the S-plane rather than the Z-plane).

FIGURE 1.5.6.1 A/D AND D/A CONVERTERS

ANALOG WORLD REQUIRES INTERFACE WITH MICROCOMPUTERS

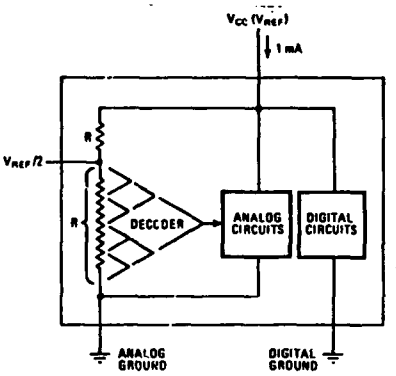
● BECKMAN 7581C	DAC	12 BIT	$24.50 EACH	CMOS
● HYBRID SYSTEMS DAC9331-16	ADC	16 BIT	$99.(2 CHIPS)	HYBRID
● DATEL/INTERSIL ADC-815	ADC	8 BIT	$205.	HYBRID
● ANALOG DEVICES A-D 558	DAC	8 BIT	$5.95	I^2L
● SIEMANS SDA 510	DAC	6 BIT (10 NS)		ECL
● NATIONAL ADC0801	ADC	8 BIT		C MOS

CHARGE BALANCED COMPARATOR

Linearity	1/4 LSB
Conversion time (f_{clk} = 640 kHz)	100 μs
Analog input range	0-Vcc
f_{clk} range (external RC)	100 – 800 kHz
Supply voltage range	4.5 – 6.3 Vdc
Current drain	1.8 mA dc
$V_{REF/2}$ range	0 – 5 V dc
Logic input levels	Std TTL
Logic output	1 TTL load
Package	20-pin dual in-line

120 x 127 MIL CHIP CONTAINS:

- 8 RESISTORS
- 4 CAPACITORS
-32 SWITCHES



ADJUSTMENT FOR TRANSDUCERS

FIGURE 1.5.6.2

A/D & D/A CONVERTERS

● ECL PROVIDES 10 NS MONOLITHIC DAC

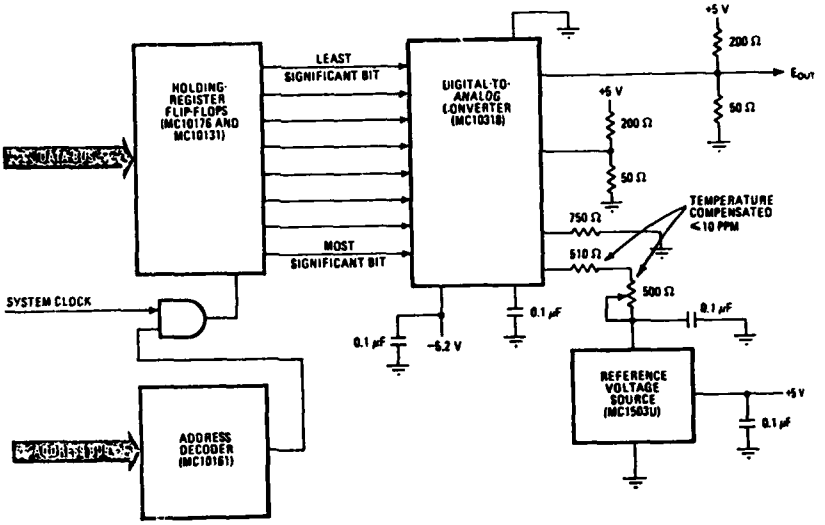
MOTOROLA MC10318

WORKS WITH

ECL BIT SLICE

MICROPROCESSOR

10800 FAMILY



● 18 BIT DAC HYBRID SYSTEMS DAC 370-18

-50MW POWER

-$210 COMM'L/$470 MIL TEMP

-2 μSEC CONVERSION

-HYBRID TECHNOLOGY

NO OUTPUT OP AMP

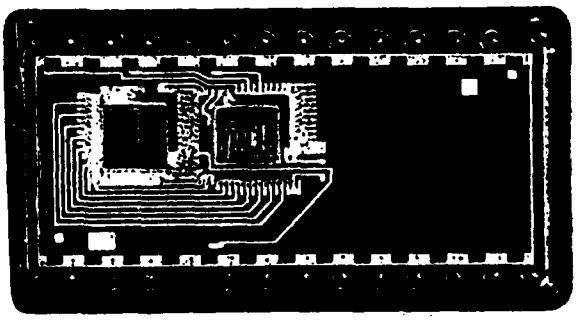
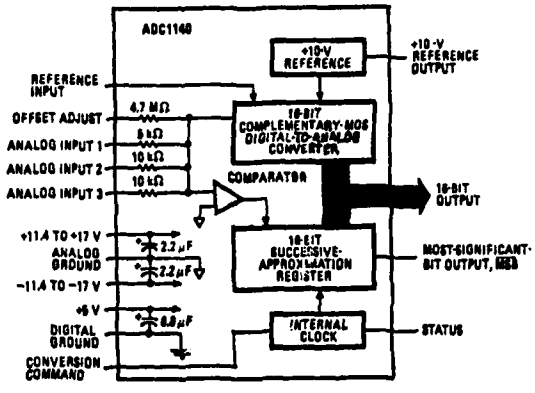
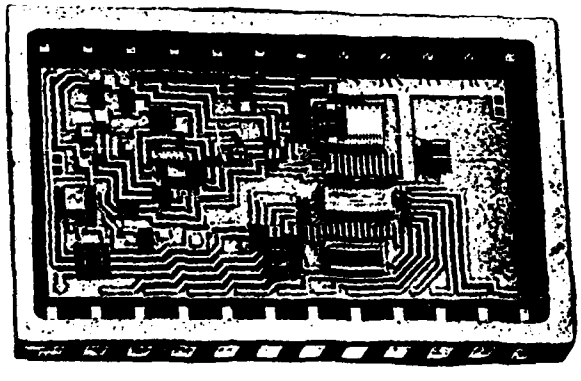


FIGURE 1.5.6.3 16 bit ADC CONVERTS IN 35 MICROSEC.



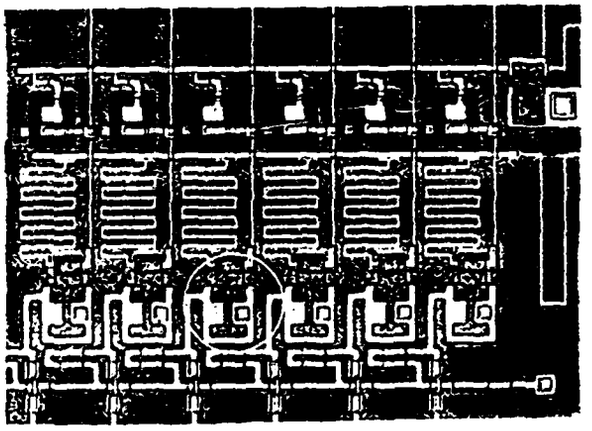
- ADC 1140
- $199/unit
- ANALOG DEVICES, INC.
- 1.2 W Dissipation
- CMOS Design
- -25°C to 85°C

12 bit DAC Settles in 35 Nanosec. FIGURE 1.5.6.4



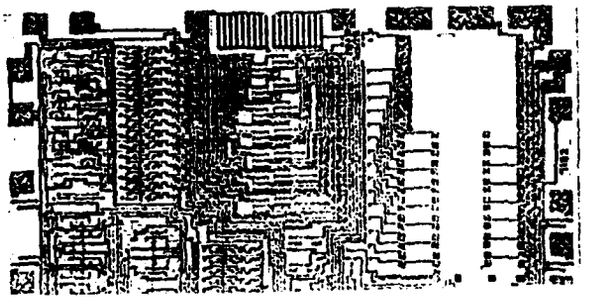
DAC 63BG
Burr-Brown
+/-.012% Linearity
960MW
$108

FIGURE 1.5.6.5 SUPERFAST ADC - 500 PICOSEC CONVERSION



NBS JOSEPHSON JUNCTION ADC
CRYOGENICALLY COOLED
6 BIT RESOLUTION
2 BILLION SAMPLES/SEC
SUPER-COOLED QUANTUM INTERFERENCE DEVICES (SQUID'S)

D-a converter chip is linear to 14 bits



HYBRID SYSTEMS HS3140
$33 EACH COMM'L $79 MIL 833B
AC OR DC REF VOLTAGE

FIGURE 1.5.6.6
ANALOG MICROCOMPUTER (8 BIT BASED)

- INTEL 2920 ANALOG-IN/ANALOG-OUT
  - ON CHIP ADC AND DAC
  - PIPELINE DIGITAL PROCESSOR
  - RAM FOR DATA
  - EPROM FOR PROGRAM CODE
  - TYPICAL APPLICATIONS INCLUDE:
    - Speech Synthesis
    - Filters
    - Modems
    - Tone Receivers
    - Mixers

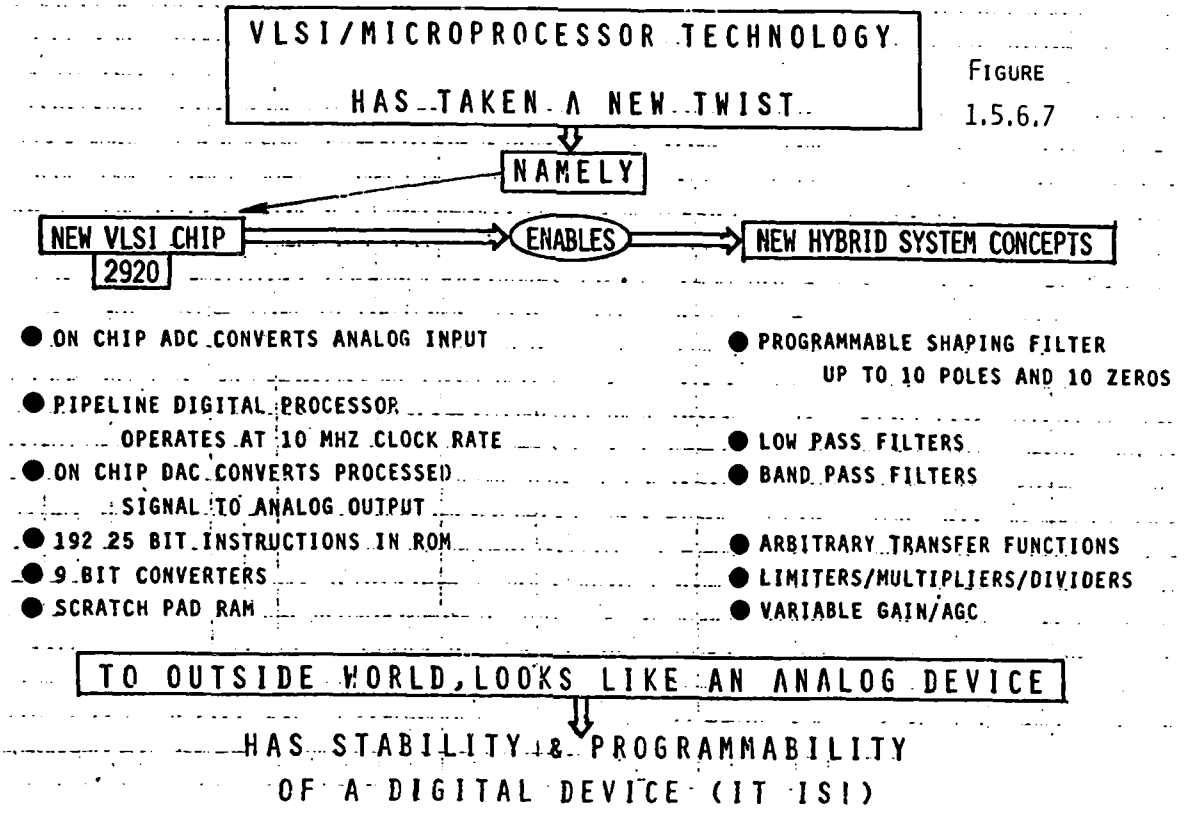
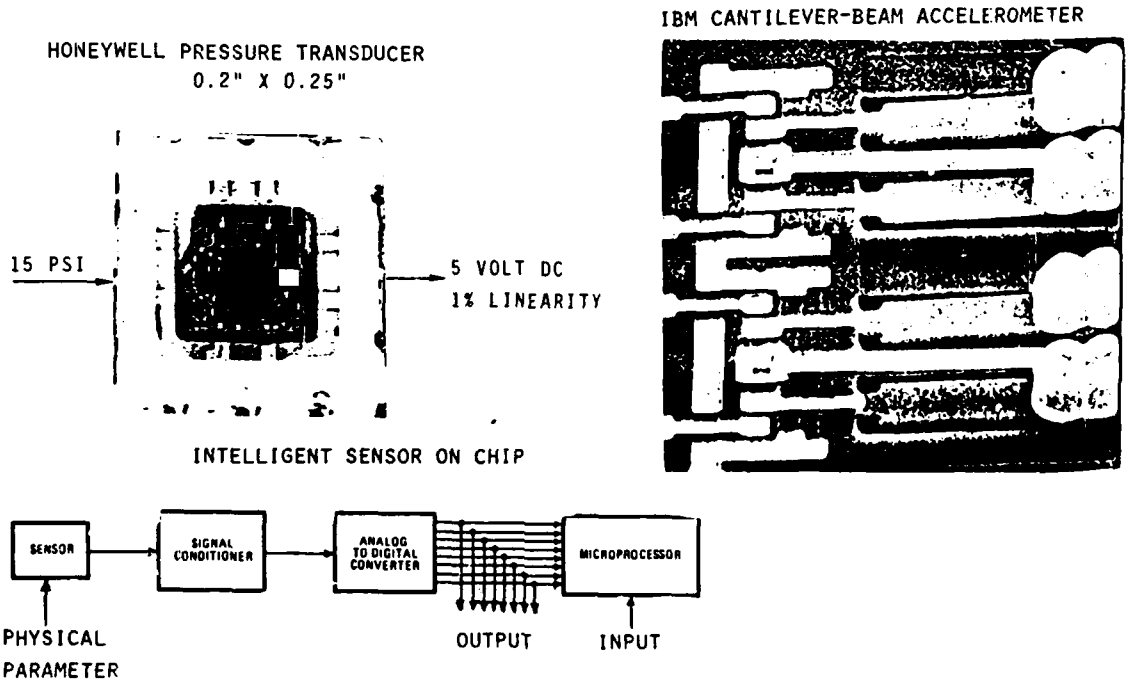


FIGURE 1.5.6.7

1.5.7 Integrated Sensors

An important trend in sensors is the integration of sensors on silicon chips which incorporate signal conditioning ADC and possibly a microprocessor on the same chip as the sensor as shown in figure 1.5.7.1. A silicon pressure transducer which consists of a diaphragm etched into the silicon chip with the piezo resistive elements diffused on each side of the diaphragm and connected into a wheatstone bridge level from a few millivolts to 5V DC with a linearity of about 1%.

FIGURE 1.5.7.1 MONOLITHIC SILICON SENSORS



A second sensor shown in the figure is a four beam accelerometer fabricated by IBM, which operates with a resistive bridge circuit similar to the pressure transducer. An accelerometer orthogonal to the plane of the paper causes a deflection of the cantilever beams proportional to the applied acceleration. A DC voltage output proportional to acceleration is produced.

Another solid state sensor of importance to guidance and control applications compatible with microprocessor technology is shown in figure 1.5.7.2. The sensor is a fibre optics gyr consisting of two fibre optics coils wrapped on a spool. This provides two separate long optical path lengths (e.g. 1000 meters each) with lasers to provide counter rotating beams detectable by electro-optical detectors. The problem of lock-up present in the conventional laser gyro is eliminated in the fibre optics gyro because the counter rotating beams are in separate fibre optic paths. An experimental gyro fabricated by Siemans of the Federal Republic of Germany has a drift rate which supports an IMU with 0.5NMi/hour accuracy.

The projected drift rate for this type of instrument is .001 degrees per hour, which will support a 0.05NMi/hour accuracy.

Lear Siegler - Grand Rapids and Hughes under contract to the Jet Propulsion Laboratory are developing such a gyro in addition to the companies listed on the figure. The Highes/JPL gyro will have 2km path lengths.

FIGURE 1.5.7.2
FIBRE OPTICS Solid State Gyro

SIEMANS

- Drift Rate of .01deg/Hr 0.5NMi/Hr Inertial Quality
- Experimental Gyro - 1000M Fibre Optics Coil
  - 12cm Diameter
- Projected Size 10cm Diameter x 3cm High
- Projected Drift Rate .001deg/Hr 0.05NMi/Hr

Others in the Race

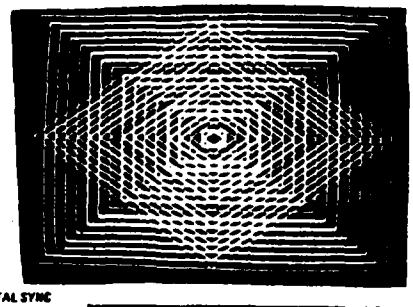
- AEG Telefunken
- Rockwell
- Honeywell
- Thomson-CSF
- NOSC (US Navy)

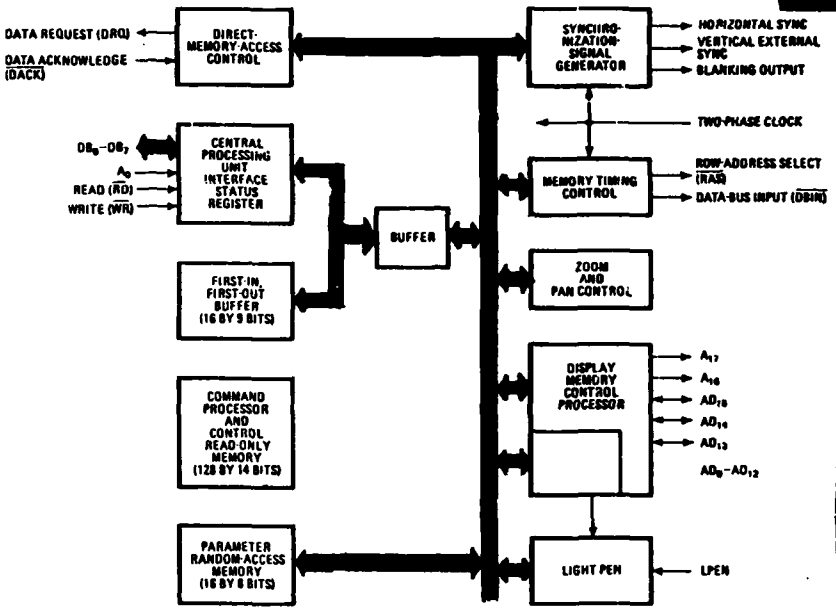
1.5.8 Color Graphics Controller

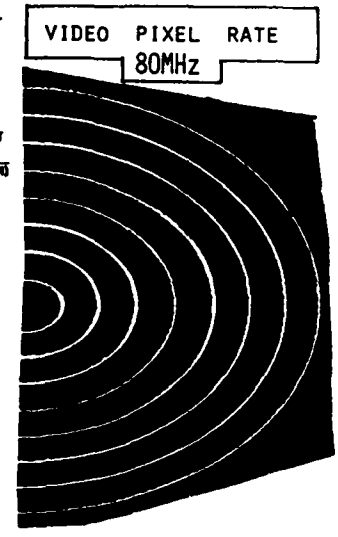
Color graphics capability is provided by a single chip peripheral controller as shown in figure 1.5.8.1. The Nippon Electric micro PD 7220 graphics controller chip provides a 2K by 2K dot matrix with full color overlays. The chip permits the generation of geometrical figures as shown in the figure. NEC has fabricated sample quantities as of late 1981.

FIGURE 1.5.8.1 HIGH RESOLUTION COLOR GRAPHICS

- μPD 7220 GRAPHICS CONTROLLER
  - -- 2K x 2K DOT MATRIX
  - -- FULL COLOR OVERLAYS RED/GREEN/BLUE
  - -- GENERATES GEOMETRICAL FIGURES
  - -- 3 MICRON/N-MOS 13,000 TRANSISTORS







1.6.0 VHSI/VLSI TECHNOLOGY

The DoD very high speed integration (VHSI) program has been introduced to complement the very large scale integration (VLSI) technology under development by commercial industry.

1.6.1 VHSI Program Goals and Objectives

The DoD sponsored VHSI program goals are summarized in figure 1.6.1.1. The VHSI program has adopted a figure of merit called functional throughput rate (FTR), defined as the product of gates per chip times clock rate. The goal is to achieve an FTR of 10^{14} gate-hertz (upgraded from 10^{13} gate-hertz) during the course of the program.

A major thrust of the $210 million program is to achieve sub-micron lithography with emphasis on Silicon, bi-polar, and MOS technology. GaAs technology has been excluded from the program, however, it may be included later. The VHSI schedule is shown in figure 1.6.1.2, with the goal of pilot production of sub-micron chips by 1986. Considering Bell Laboratories has already achieved sub-micron experimental chips of 0.3 micron, this goal does not seem too difficult.

The VHSI goal is to achieve a throughput of 1 million operations per second. Again, the Intel 432 has achieved a throughput of 0.5 million operations per second already. The VHSI goal of achieving a FTR of 10^{14} gate-hertz by 1986 appears reasonable since the Hewlett Packard 32 bit microprocessor has achieved a gate-hertz of 1.35×10^{12} in 1981 using 1.5 micron lithography.

The Bell Lab 0.3 micron Si MOSFET which operates at a clock rate of 10gHz should be able to achieve 450,000 gates which would yield an FTR of 4.5×10^{15}. Bell Labs has the capability of achieving this FTR within a short time, certainly before the 1986 goal of the VHSI program.

The DoD VHSI program estimates for performance for various military functions is shown in figure 1.6.1.3.

The prime contractors and team members for the phase I of the VHSI program are listed in figure 1.6.1.4 together with the FTR requirments for defense electronics needs.

The annual funding of the VHSI program is about $36 million, approximately 7% of United States Industry's micro-electronics RD expenditure level. DoD hopes to focus some short-supply micro-circuit designer's efforts on DoD's unique needs.

FIGURE 1.6.1.1. VHSI/VLSI TECHNOLOGY

- VHSIC PROGRAM GOALS - DoD SPONSORED $210M FUNDING
  - -PILOT PRODUCTION OF 250,000 GATE PROCESSORS BY 1986
  - -25 MHz CLOCK
  - -10^6 TO 3 x 10^9 OPS/SEC
  - -SUBMICRON LITHOGRAPHY 0.5 TO 0.8 MICRON
  - -SILICON BI-POLAR AND MOS TECHNOLOGY
  - -FTR (FUNCTIONAL THROUGHPUT RATE) = GATES/CHIP x CLOCK RATE
    10^{13} GATE-HERTZ
- VHSIC APPLICATIONS

Sonar—Acoustic signature analyzers used in the BQQ-5 and BQQ-6 processing subsystems in strategic (Trident) and attack submarines; the MK-48 processor in homing torpedoes; and in the Proteus processor used in antisubmarine warfare aircraft.

Radar—Signal processors for radar systems in the E2C and E3A airborne early warning systems; in advanced fighters (F-14, F-15, F-18) and interdiction aircraft (A-6) for all weather bombing; and in stand-off target acquisition systems (SOTAS).

Missile Guidance Satellites—Processors of radar and infrared sensor data for inertial navigation (Global Position Satellite), and processors for target recognition, proximity fusing, and clutter rejection in air-to-air missiles (Phoenix, Sparrow, Sidewinder), surface-to-air missiles (Patriot, Hawk), and submarine-launched cruise missiles.

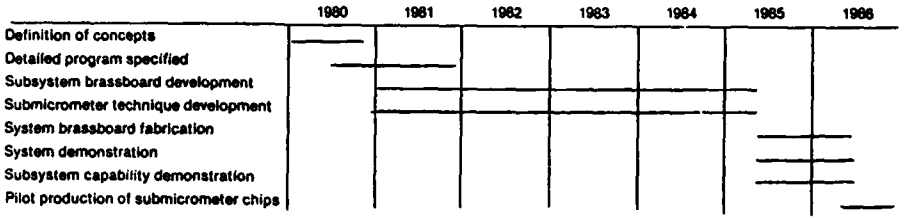
Communications—Spread spectrum and time dispersion modulators and demodulators, error correction coders and decoders for digital voice transmission (ANDVT) and battlefield communications (REMBASS, SeekTalk, SINGGARS).

Signal Intercept Analysis—Signal modulation analyzers and signal classifiers for scan receivers (ALR-66, ALR-67)

Electro-Optical Processors—Processors of electro-optical data for more detail and for estimation of target trajectories, in such infrared surveillance systems as the Halo satellite

VHSI/VLSI TECHNOLOGY (CONT.) FIGURE 1.6.1.2

- VHSIC SCHEDULE



- GROWTH IN SYSTEM PROCESSING

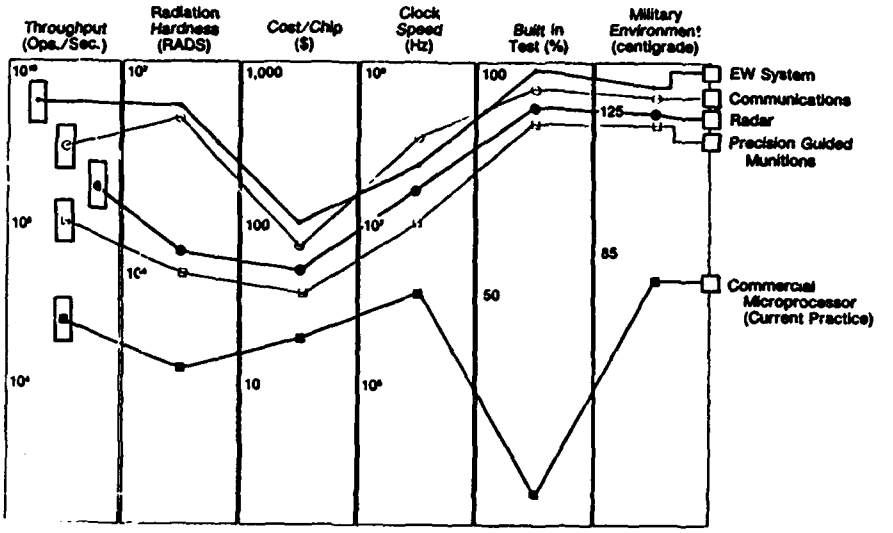
Application	Platform	Equivalent signal processing required, million instructions per second: Current	Future
Army tactical signal intelligence	Land based mobile	0.4	40
Cruise missile terminal guidance	Small missile	0.1	50
Data correlation for over-the-horizon fire control	Ship/land based	1	50–100
Airborne synthetic aperture radar	Aircraft/spacecraft	3	100–500
Electronic warfare radar pulse processor	Aircraft/spacecraft	2	200–300
AJ/LPI spread spectrum communications	Small missile/RPV aircraft/spacecraft	5	500
Wideband data links 1 Gb/s	Spacecraft	10	500
U/SEA global search (SOSUS)	Ship/land based	0.5	2 000
ELINT/ESM processor (10 Hz digital spectrum analyzer)	Aircraft/spacecraft	10	10 000

- GALLIUM ARSENIDE VLSI BY 1985 (NOT PART OF DoD VHSIC PROGRAM)
  - -COMPLEMENTARY TO Si FOR GIGABIT/GIGAHERTZ APPLICATIONS
  - -STRATEGY MAY BE TO USE GaAs AS FRONT END PROCESSORS TO SLOW SIGNALS TO SILICON SPEEDS
  - -GaAs CONSIDERED FOR VHSIC II (LATER)

US DoD VHSI PROGRAM

FIGURE 1.6.1.3

DoD PERCEIVES THEIR REQUIREMENTS EXCEED COMMERCIAL MICROPROCESSOR CAPABILITIES



- $225M FUNDING OVER NEXT 6 YEARS EXPECTED TO CHANGE THIS
- 9 CONTRACTORS FINISHED PHASE ZERO STUDIES
- 3 CONTRACTORS EXPECTED FOR PHASE ONE DESIGN

FIGURE 1.6.1.4 US DOD'S VHSI CIRCUIT PROGRAM UNDERWAY

Prime contractor & team members	Phase 1: (Proposal) Technology	Phase 1: (Proposal) Architectural Approach	Phase Zero Monitor
General Electric Intersil Martin Marietta* Analog Devices Tektronix Stanford Univ.	C-MOS and C-MOS/SOS	Standard chip set of seven logic devices, three memory chips	Army
Honeywell	Bipolar (integrated Schottky logic, current mode logic, and current sourcing logic)	Standard families of chip sets and CAD	USAF
Hughes Aircraft Signetics Burroughs Res. Triangle Instit. Cornell Univ. Stanford Univ. UCLA	C-MOS/SOS and bipolar (integrated) Schottky logic).	Not available	Army
IBM Northrop†	N-MOS	Standard chip set for data processing and macro cell standard custom via CAD	Navy
Raytheon Fairchild Semiconductor Varian/Extrion	Bipolar (isolated integrated injected logic)	Standard chip set of four and custom "glue chips" using gate arrays	USAF
Rockwell Sanders Assoc. Perkin-Elmer	C-MOS and C-MOS/SOS	Standard chip set: four logic plus two memory	Army
Texas Instruments	Bipolar (Schottky transistor logic) N-MOS memory	Standard chip set of five and gate array "glue chips"	USAF
TRW Motorola Sperry Univac GCA-Mann	Bipolar (three-diffusion) and C-MOS	15-20 chips total in two technologies	Navy
Westinghouse National Semiconductor Control Data Corp. Harris Electronics Mellon Institute Boeing Aerospace	C-MOS and C-MOS/SOS	Standard chip set of six and set of gate array "glue chips".	Navy

DEFENSE NEEDS

APPLICATION	THROUGHPUT RATES (GATE Hz)
COMMAND, CONTROL, AND COMMUNICATIONS	10^{13}
RADAR	10^{14}
ELECTRONIC WARFARE	10^{14}
IMAGE PROCESSORS	10^{13}
COMPUTERS	10^{13}
ANTISUBMARINE WARFARE	10^{14}

VHSIC CAPABILITIES

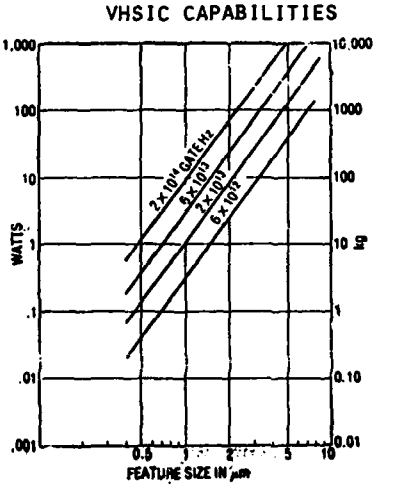


FIGURE OF MERIT: THRUPUT (HZ) X NO. GATES ON VHSI CHIP

ANNUAL FUNDING DOD

$36M 7% OF US INDUSTRY MICROELECTRONICS RESEARCH

1.6.2 VLSI University Research Centers

An important VLSI program is the DARPA cooperative university program to promote the training of students to design ultra-complex VLSI circuits using computer aided design for fast turn-around of chip fabrication. The participating universities are listed in figure 1.6.2.1, with Professor Carver Mead of CalTech as the unofficial ring leader of a distinguished group of university professors.

This program promises a big pay-off during the 1980's -- namely the development of a cadre of university graduates who can design custom VLSI chips with short turn-around.

FIGURE 1.6.2.1 DARPA FUNDS US. UNIVERSITIES VLSI RESEARCH CENTERS

COMPUTER AIDED DESIGN
FAST TURNAROUND CHIP FABRICATION
SYSTEMS ARCHITECUTRE

FUNDING		
1979	1980	1981
$2M	$7M	$10M

TRAIN STUDENTS FOR DESIGN & APPLICATION OF ULTRACOMPLEX VLSI CIRCUITS

⇓

Big Pay-Off For 1980's

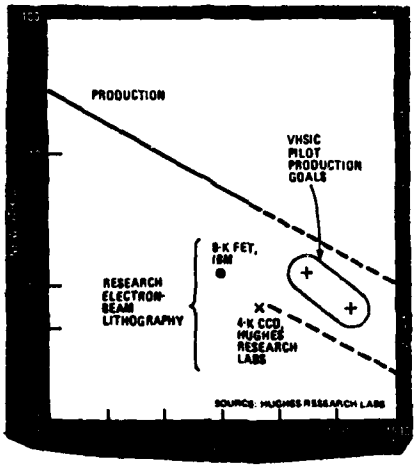
PARTICIPATING UNIVERSITIES

CALTECH	PROF. CARVER MEAD
CARNEGIE-MELLON	PROF. DAVID LEWIN
M.I.T.	PROF. PAUL PENFIELD, JR.
STANFORD	PROF. JOHN LINVILL
U.S.C./I.S.I	PROF. JOHN NELSON
UNIV. OF CAL. BERKLEY	PROF. DAVID HODGES
CORNELL	PROF. EDWARD WOLF (NSF FUNDING)

1.6.3 VLSI Trends for the Future

The reduction in circuit feature size resulting from submicron lithography will have dramatic impacts upon circuit performance. For example, a reduction from 4 micron to 1 micron results in a delay-power product reduction of 64 and a density increase of 16 times. A reduction from 1 micron to 0.25 micron would likewise result in the same improvements. The impact on circuit parameters is summarized in figure 1.6.3.1.

FIGURE 1.6.3.1



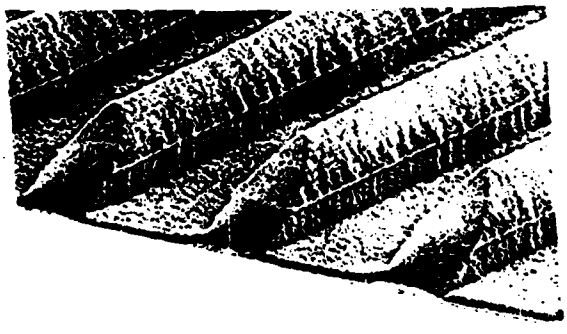
1 MICRON LINES

POLYSILICON, PLASMA-ETCHED

o SUBMICRON SCALING FACTORS

4 MICRON TO 1 MICRON

Parameter	Change with linear scaling, factor of 4*
Circuit density	16
Operating voltage	1/4
Device current	1/4
Power per circuit	1/16
Circuit capacitance	1/4
Delay per circuit	1/4
Line resistance	4
Line response time	1



1.7.0 HIGH LEVEL LANGUAGES (HLL) FOR MICROPROCESSORS

1.7.1 HLL Trends

The use of HLL dominates the trend in microprocessor software. The promise of HLLs to reduce software development, debugging, and maintenance should result in a significant reduction in life cycle costs. The savings are off-set by penalties in memory and speed which result from the use of HLL coding.

The HLL trends are summarized in figure 1.7.1.1. The new DoD standard HLL Ada was adpted in late 1980. The availability of Ada compilers appears to be occuring sooner than originally expected. The entry of Intel into the 32 bit microprocessor market with an Ada compiler is accelerating the trend to Ada.

FIGURE 1.7.1.1

MICROPROCESSOR SOFTWARE TRENDS

HLL DOMINATE TREND

- PASCAL INCREASING IN POPULARITY
  - -ELEGANT, ENGLISH-LIKE LANGUAGE
  - -MICRO CONCURRENT PASCAL BY ENERTEC FOR SEVERAL MICROPROCESSORS
    - HANDLES EVENTS AND SYNCHRONIZES TASKS FOR REAL TIME PROGRAMS
  - -HP 64,000 CROSS COMPILER TARGETED TO 8080/8085, $2000 COST
  - -STANDARDIZATION PURSUED BY ANSI X3J9, NBS, AND IEEE COMPUTER SOCIETY
- DoD NEW ADA FINALIZED AUGUST 1980
  - -INPUT/OUTPUT
  - -OPERATING SYSTEM FEATURES SUCH AS RUN TIME SOFTWARE

NEED ADA COMPILERS FOR POPULAR MICROPROCESSORS

  - -PROVIDES MAXIMUM UNIFORMITY FOR MYRIAD OF DoD COMPUTERS
- HLLs WIDELY AVAILABLE FOR MICROPROCESSORS
  - -FORTRAN
  - -COBOL
  - -PL/1

HLL PUSH

REDUCE SOFTWARE DEBUGGING AND MAINTENANCE

LIFE CYCLE COST SAVINGS OFF-SET MEMORY/SPEED PENALTIES

- SOFTWARE COSTS
  - -$10/LINE OF VALIDATED CODE
  - -30% DEVELOPMENT, 70% MAINTENANCE & DEBUGGING

1.7.2 Available HLL Compilers.

A number of HLL compilers are available for the popular microprocessors as is summarized in figure 1.7.2.1. FORTRAN is still a widely used language for guidance and control applications. For example, NASA Dryden Flight Research Center uses FORTRAN for all of their real-time control law computations in their research RPV (remotely piloted vehicle) guidance and control systems. FORTRAN compilers are available for most of the popular microprocessors.

PASCAL is another popular HLL well suited to modern structured program. TI has adopted PASCAL as its corporate language, USC has adopted PASCAL as the standard language for computer science students. PASCAL compilers are available for most of the popular microprocessors including the mature 16 bit microprocessors (8086, Z8000, and 68000).

JOVIAL, the USAF standard HLL defined in MIL-STD-1789 is an excellent HLL language suitable for avionics. JOVIAL is similar to HAL/S, the high-level avionics language used for the space shuttle avionics. GDE is developing a JOVIAL compiler for the Z8000 microprocessor.

FORTH is a unique block structured language suitable for real-time systems. FORTH is available for most microprocessors. FORTH users are enthusiastic about its capabilities.

FIGURE 1.7.2.1

- HIGH LEVEL LANGUAGES WIDELY AVAILABLE FOR MICROPROCESSORS
  - FORTRAN COMPILERS
    -8080/8085/8086/Z-80
    -LSI 11/23
    -6800/6502
    -Z-80/Z8000
  - PASCAL COMPILERS
    -8080/8085/8086
    -6800/6502/68,000
    -Z-80, 8000
  - JOVIAL COMPILER
    -Z8001 GDE DEVELOPMENT
  - ADA (NEW DoD LANGUAGE)
    ADA for iAPX432 is expected in early 1982
  - FORTH
    -AVAILABLE FOR MOST MICROPROCESSORS
    _A UNIQUE HLL, BLOCK STRUCTURED LANGUAGE
  - PL/1 DERIVATIVES
    -PL/M FOR 8080/8085/8086
    -PL/Z FOR Z80/Z8000

Intel has pushed adaptations of PL/1 for its microprocessors, with versions of PL/M avaliable for the 8080, 8085, and 8086. Zilog has a version called PL/Z for the Z80 and Z8000. PL/1 has draw-backs for use in real-time systems. Honeywell considered using PL/M for its 8086 based Digital Advanced Avionics System (DAAS) developed for NASA Ames, but rejected it as being unsuitable.

Ada is the new DoD Standard HLL (see figure 1.7.2.2). Considerable compiler activity is underway and is summarized in the figure. Intel is scheduled to have an Ada compiler for its new iAPX-432, the 32 bit microprocessor in early 1982.

AUGUSTA ADA BYRON
THE FIRST PROGRAMMER
1816-1852

FIGURE 1.7.2.2

Ada

DoD STANDARD HIGH-LEVEL LANGUAGE
ADOPTED DECEMBER, 1980
ANSI / ISO ADOPTION 2-3 YEARS AWAY

MUCH COMPILER ACTIVITY UNDERWAY

- COMPILER FOR WESTERN DIGITAL MICROENGINE --- EARLY 1981
- INTEL iAPX432 ADA COMPILER PLANNED / FIRST ADA BASED MICROPROCESSOR
- SOFTECH COMPILER FOR VAX 11/780 --- LATE 1981/EARLY 1982
- ADA COMPILERS FOR IBM COMPUTERS UNDER USAF CONTRACTS TO
  - TI
  - INTERMETRICS
  - CSC

Figure 1.7.2.3 summarizes a PASCAL cross compiler for the 8086 microprocessor hosted on the PDP-11. This cross compiler called PASPORT was designed by Intermetrics, the company which developed the HAL/S compilers for the Space Shuttle.

FIGURE 1.7.2.3 PASCAL CROSS-COMPILER FOR 8086

INTERMETRICS PASPORT $15,000

HOST PDP-11

OBJECT 8086 MICROPROCESSOR

Object Code Transfer via RS-232C

1.8.0 SUMMARY AND CONCLUSIONS

The summary and conclusions are set forth in the tables below.

SUMMARY & CONCLUSIONS

COMMERCIAL VLSI DEVELOPMENTS ACCELERATING

- NEW 32 BIT HIGH-PERFORMANCE MICROCOMPUTERS WILL HAVE DRAMATIC IMPACT
- NEW HIGH-PERFORMANCE 16 BIT PROCESSORS BECOMING AVAILABLE
- RECENT 16 BIT MICROPROCESSORS ARE MATURING
  - -- 8086/68,000/Z8000
  - -- SUPPORT SOFTWARE AVAILABLE
- SILICON TECHNOLOGY EXPECTED TO REMAIN DOMINANT
  - -- N-MOS STILL MOST USED
  - -- C-MOS CONVERSIONS INCREASING
  - -- CMOS/SOS BOOSTED BY TOSHIBA/ROCKWELL DEVELOPMENTS
- 1-2 MICRON LITHOGRAPHY NOW BEING USED

NEW 0.3 MICRON X-RAY LITHOGRAPHY USING Si MOSFETS YIELDS
- -- SWITCHING SPEED 30 PICO SEC
- -- SPEED-POWER PRODUCT OF 5 FEMPTOJOULES

DoD VHSI PROGRAM UNDERWAY

- WILL ACHIEVE SUBMICRON LITHOGRAPHY WITHIN 5 TO 6 YEARS
- WILL ACHIEVE 10^{14} HERTZ-GATES PER CHIP BY 1986
- WILL ENCOURAGE SEMI CONDUCTOR COMPANIES TO DEVELOP DoD REQUIRED DEVICES

THRUST TO HIGH LEVEL LANGUAGES CONTINUING

- ADA COMPILERS DEVELOPING FASTER THAN EXPECTED
- PASCAL IS WIDELY AVAILABLE AND USED FOR MICROPROCESSORS
- NEW ARCHITECTURES ENCOURAGE HLL

IMPACT OF DoD STANDARDIZATION BEING FELT

- INSTRUCTION SET ARCHITECTURE STANDARD DESIRED
  - BY USAF (ASD AVIONICS CONTROL) AND NAVY
  - --- GD FW PLANNING MIL-STD-1750 MICROPROCESSOR CARD FOR F16
  - --- ZILOG MAY PRODUCE MIL-STD-1750 MICROPROCESSOR
- DoD SERVICE HLL STANDARDS (E.G. USAF JOVIAL) TO BE REPLACED BY ADA IN FUTURE
  - ---USAF HAS TRANSITION PLAN
  - --- PROGRAMMING IN ADA SOURCE
  - --- TRANSLATE ADA TO JOVIAL SOURCE DURING TRANSITION
- SERIAL DATA BUS MIL-STD-1553B
  - ---ENHANCED BY VLSI BIU CHIPS
  - ---COMPATIBLE WITH FIBRE OPTICS BUS
  - ---ARINC STDS (E.G. 429) PROLIFERATES BUS PROTOCOLS

MEMORY IMPROVEMENTS CONTINUE

- 64K RAM's MATURING
  --- ALPHA PARTICLE ERRORS DIMINISHED BY DIE COATING
- 128K ROMs EXPECTED NEXT YEAR
- 16K EE PROMS AVAILABLE
- RAMs WITH LOW POWER KEEP-ALIVE POWER
  --- N-MOS AND C-MOS RAMs PROVIDE PSEUDO-NON VOLATILE MEMORY WITH BATTERIES
- BUBBLE MEMORIES STILL EXPENSIVE
  --- ROCKWELL HAS DROPPED COMMERCIAL MARKET
  --- IBM CONTIGUOUS-ELEMENT LOOKS PROMISING
  --- 1M BIT AVAILABLE / 4M BIT COMING
  --- TI HAS DROPPED OUT

SPECIAL PERIPHERAL CHIPS INCREASE

- ANALOG INTERFACE TECHNOLOGY CONTINUES IMPROVEMENT
  --- ADC CHIPS AVAILABLE, LINEARITY IMPROVING
  --- DAC CHIPS LO COST AND FAST
  --- 500 PICO SEC/6 BIT ADC ACHIEVED
- SPEECH GENERATION CHIPS NOW AVAILABLE AND EASY TO APPLY
- SPEECH RECOGNITION CHIPS BECOMING AVAILABLE
  --- VERY LIMITED VOCABULARY 10-25 WORDS
  --- GENERAL RECOGNITION PROBLEM VERY COMPLEX
- NEW ARITHMETIC PROCESSORS AND SPECIAL I/O CHIPS PROLIFERATE

CHAPTER 2

MICROPROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL ARCHITECTURES

by

Richard F. BOUSLEY
Boeing Military Airplane Company
P.O. Box 3707, M/S 41-08
Seattle, Washington 98124

U S A

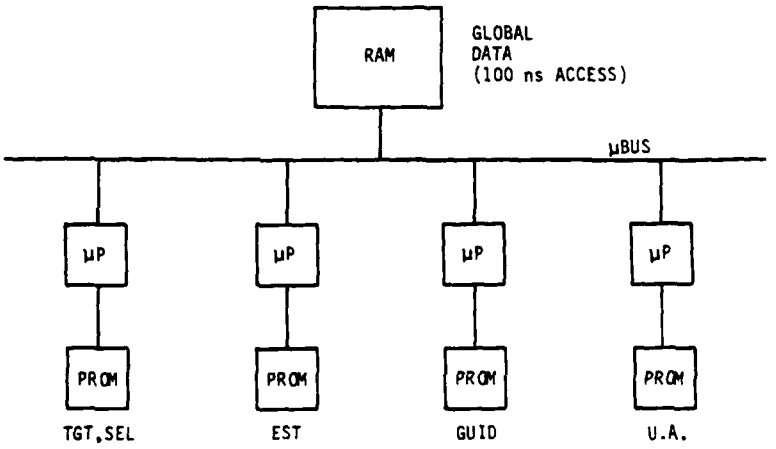
2.1.0 INTRODUCTION

2.1.1 BACKGROUND

Control of modern day jet-powered aircraft, particularly tactical aircraft, imposes a tremendous workload on the pilot and/or crew. The number of tasks required by a fighter pilot, for example, to fly the aircraft, navigate, maintain target and hostile aircraft surveillance, and still monitor instruments stretches the abilities of even the most skilled fliers.

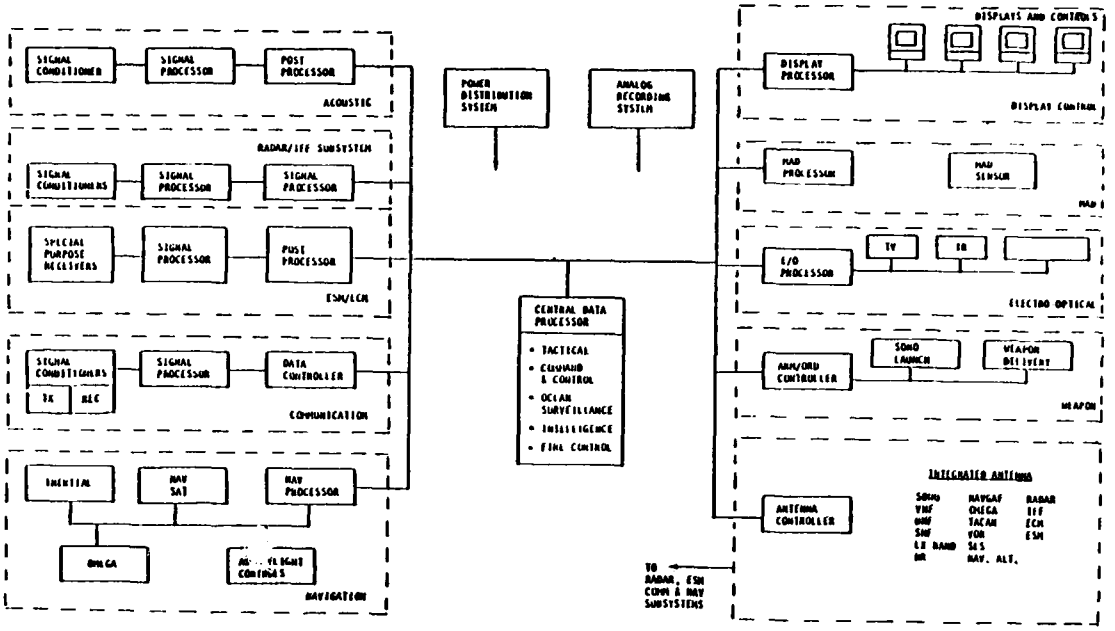
When the first transistorized computers became available, many of the less complex and repetitious control tasks were relegated to a centralized minicomputer. An example of a system employing a centralized mini computer is the USN F14. The vulnerability of a centralized architecture was soon recognized by avionics designers, who began to consider the use of federated architectures. Within a federated architecture, functions are serviced by dedicated processors which communicate with a "master" via a data bus. The master processor is generally a minicomputer with requirements for a large mass memory to be appended to it, allowing most information processing functions to be performed in this processor. Federated architectures are less vulnerable to failure than centralized architectures, but the concept of a single master processor limits the overall fault tolerance that can be achieved.

The need for fault tolerance is pushing investigation of advanced architectures which employ a "shared memory concept". Shared memory permits satellite or remote processors to directly access main memory. Thus, if a processor is lost due to any reason, its functions can be reallocated to another processor and a mission can be continued. This concept is illustrated below:



A good example of a federated architecture is illustrated in the block diagram in figure 2.1, which highlights the master central data processor and the single federated busing structure. This configuration has been a mainstay for most modern-day military aircraft G&C system architecture design studies. This type of architecture was a fairly reliable and efficient means to connect and provide communication to the computational elements of the day. However, the advent of flight-qualified microprocessors is affording the avionics system designer the opportunity to implement new architectures and architectural concepts heretofore only explored in paper studies or in large-scale laboratory simulations.

FIGURE 2.1 SURVEILLANCE - ATTACK AIRCRAFT FEDERATED ARCHITECTURE



Preliminary investigation seems to point to the fact that the major impact of microprocessors in G&C architectures is that microprocessors will provide the ability for the system designer to construct highly distributed and consequently extremely complex system architectures. The real issue seems to be how to limit the system designer but not adversely affect the potential performance of the G&C system architecture.

2.1.2 OBJECTIVES

The objective of this chapter is to present a sample of techniques now being examined for use of microprocessor hardware and software in future military G&C systems. Many ongoing related investigations are being conducted by industry and the military, into the use of microprocessors in distributed architectures to obtain increased G&C systems performance.

The introduction of microprocessors into distributed G&C systems demands that care be taken to avoid potential problems relating to hardware and software proliferation and increased maintenance logistics support. Although microprocessor hardware is relatively inexpensive, the associated operational and support software is not. The multitude of microprocessor types available today, each having their own support software, could compound the logistic and acquisition problems if proper management controls are not utilized.

These potential problems can be possibly circumvented through the proper application of standards to minimize the proliferation of microprocessor types, languages, and development systems. In addition, the selection of a microprocessor type could consider a broad base of G&C applications so that a wide range of G&C applications could be accommodated. Technology independence must also be considered in order to extend the period of time during which some selected standard would be applicable.

The major impact of microprocessors upon G&C system design is the degree of distribution of the processing functions that can be achieved by the avionics systems designer. Review of the literature reveals that two basic architectural approaches are used to design distributed G&C systems. One employs redundant multiplex data buses on a single level of control and data transfer and another employs a hierarchy of multiplex data buses and levels of control and data buses.

A third type of architecture is also presented for discussion which is actually a conglomorate of the single level and hierarchial multiplex data bus systems. For this discussion it will be referred to as n-Level Multiplex Architecture. This particular type of architecture is becoming more and more prominent in the literature. The fact that this discussion is appearing in the literature implies that it is being considered by designers and consequently should be discussed and evaluated.

As stated earlier the major impact of microprocessors upon G&C system design is that, due to lower cost, weight, size, cooling, and power requirements, microprocessors will allow the system designer to utilize distributed processing schemes in virtually limitless combinations for G&C systems.

Chapter 1 of this report presented a comprehensive examination of current and projected microprocessor technology. Consequently, this chapter will discuss some current and proposed architectures that are possible because of this technology rather than discuss specific hardware or technologies.

Section 2.2 examines the structure of the basic types of architectures introduced above. Computational considerations are presented as well as software and fault tolerance considerations, for each type of architecture.

Section 2.3 presents application of these architectures to integrated G&C systems for generic types of aircraft.

Section 2.4 presents a comparison of benefits through the use of each of these types of architecture which are made possible through the use of microprocessors.

2.2.0 CANDIDATE G&C ARCHITECTURES

2.2.1 G&C SYSTEMS

A G&C system is fundamentally an application function that comprises two major elements. These elements are a complement of sensors/subsystems, which perform the search, detection, position determination, and communication functions of a mission, and a processing system fulfilling the corresponding information processing functions. Figure 2.2 characterizes the interconnection between these elements in a generic application function architecture. The interconnect topology and control structure is the primary vehicle for coordinated information flow between the processing elements and the sensor elements.

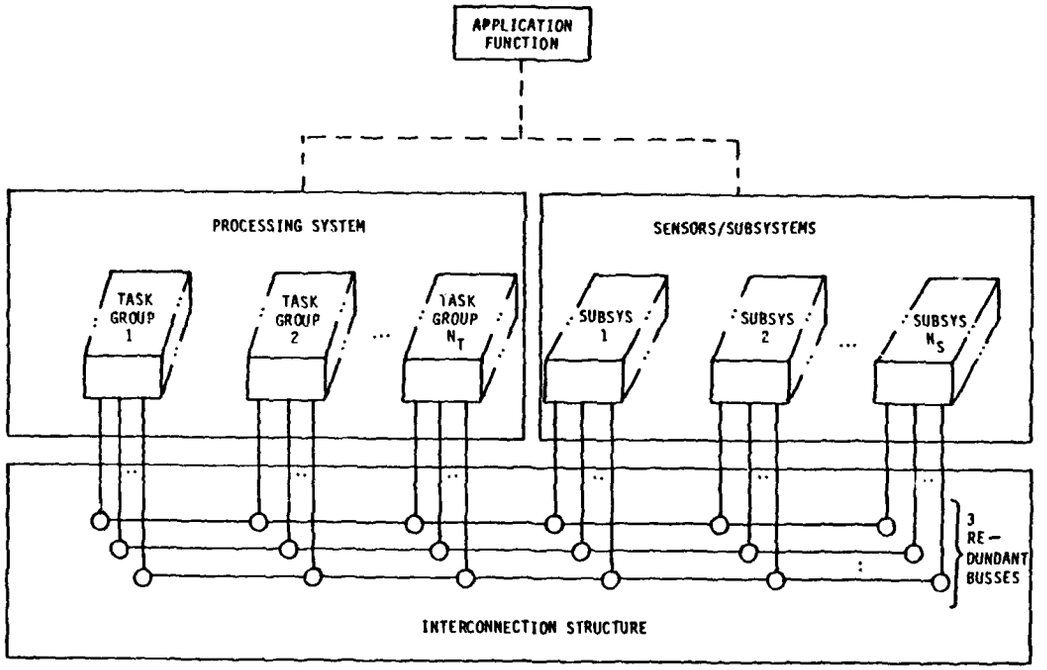


Figure 2.2 Generic Application Function Architecture

The purpose of integrated G&C systems is to reduce crew work load and provide increased performance in terms of threat detection, threat avoidance, and navigation to/from targets. A means of attaining increased performance is through enhancement of the G&C system.

A key approach in achieving increased performance is to design G&C systems that permit the crew to devote a high level of concentration to quick and precise evaluations of new situations, and to make equally quick and precise responses. The great speed at which events can unfold makes delays in response costly, if not catastrophic. To improve the quality, reliability, and timeliness of the crew performance in its mission will require the introduction of the sophisticated man-machine oriented systems. Each process must be analyzed with respect to the capability of a machine and a crew member

to perform the tasks in the time available. Such analysis may then be used to partition mission responsibilities between man and machine so that the resulting system design achieves the best capabilities of both participants. This implies that G&C systems must provide the integrating algorithms, displays, and man-machine interfaces to couple man and machine into an optimal control system to achieve the needed performance. This integration could be accomplished, if properly designed, through use of almost any modern digital processor. However, until the advent of microprocessors these types of applications were relagated to large simulation laboratories where parameters such as size, weight, power, cooling and cost were not as significant as they would be in an airborne application.

Several basic strategies are available to system architects for enhancing G&C systems:

a. Redefinition of the system requirements to increase efficiency

b. Development of functionally specific machines

c. Improvement in raw speed/storage of machines

d. Application of system and machine architectures which support concurrency

e. Combinations of the above

The success of the first two strategies depends greatly on the particular application, and the gain moderate relative to the total performance of the G&C system, because these strategies concentrate on only a part of the overall system. The third approach is limited by the state of the art in devices at that time frame. As a result, the immediate goal of most system design is the fourth strategy, which is to find methods of improving performance by creating systems which exploit parallel, concurrent, or simultaneous execution of tasks in multiple processing elements, better known as distributed systems. This strategy can be implemented through the use of microprocessors in distributed G&C architectures.

A sample architecture is depicted in Figure 2.2. This architecture employs multiple processing elements by partitioning the processing system functions into related task groups. This architecture is generic in that it exhibits multiple processing elements and an interconnect structure. Both of these items are present in all G&C architectures and are combined with a multiplex or bus control element to make up the architectures introduced earlier (i.e., single level, hierarchical, n redundant). Thus a G&C architecture comprises two elements: an interconnect structure or topology and a control mechanism or bus control. The interconnection structure provides a mechanism which permits any element to communicate with any other. With this sample architecture growth in the sensor/subsystem area to accommodate technological advances or responses to new threat conditions should only have impact on the processing system by either adding subtasks within existing task groups or by adding new task groups and making appropriate adjustments to the interconnection structure.

2.2.2 G&C SYSTEM ASSESSMENT CRITERIA

Architectures that employ multiple processors are generally classified as distributed systems. Questions pertinent to the performance and growth capability of distributed systems include the following:

- Does the system encourage modular, incremental growth and configuration flexibility of both hardware and software? - FLEXIBILITY
- Can the system easily adapt to new functions adaptability? - ADAPTABILITY
- Can the system detect, categorize, isolate, and recover from failures? - FAULT TOLERANCE
- What type of interconnect and control structure is required? - ARCHITECTURE TYPE

In response to these and similar questions regarding distributed processing architectures, this section attempts to establish general assessment criteria that can be used to trade design issues associated with the development of multiple microprocessor distributed G&C systems. The extent to which the potential benefits expected of distributed processing systems can be realized depends on the degree to which the assessment criteria are satisfied. These assessment criteria will be used in section 2.4 of this chapter to compare capabilities of distributed architecture types.

Standardization may be a desirable feature since it implies a homogeneous rather than a heterogeneous configuration but, is not really an assessment criteria. The prime advantages of standardization are interchangeability and commonality, since the numbers of different types of elements in the system can be minimized. Of course, the degree to which standardization can be incorporated is subject to various trade-studies. One important trade is how rigid the standard can be made before it actually inhibits growth and the implementation of new technology.

Another trade considers flexibility and the method for expanding the potential processing power in a system. In a multiple processor approach, expandability could be made available either by utilizing more powerful processors and reserving the extra capacity for use when adding more tasks, or by employing less powerful processors with little or no reserve capabilities and adding processors when additional tasks are desired. For complete flexibility, provisions should be made for accommodating either or both of these approaches.

Standardization trade items will be identified and examined in Chapter 4 of this report.

Other related issues concern the distribution of system-wide control and the mechanism by which the operational integrity of the system is maintained.

2.2.2.1 FLEXIBILITY

In the literature, system flexibility is referred to by various terms, such as modularity, expandability, adaptability, and extensibility. Essentially, flexibility is the ease with which system function and performance can be changed without altering the basic system design.

Although system flexibility is currently being sought using several approaches, the generic architecture in Figure 2.2 serves as a model that is suggestive of design goals for which to strive. An overall design objective may be to derive a single, versatile architecture that has the necessary features for potential application to many platforms. Toward this end, the following architectural features indicate a design philosophy that facilitates system flexibility:

- Interconnect Structure. The number of hardware resources on any bus should only be limited by the data transmission capacity of the bus.
- Task Intercommunication. Data necessary for task intercommunication should be capable of being transferred over the bus structure as a regular part of an executing operational program without degrading the overall system operation.
- Executive Control Structure. The system executive should provide an efficient means for supporting task interaction and permit ease of upgrading the applications software.

Interconnect Structure

An interconnect structure includes the bus configuration and conventions by which the various system elements are able to communicate. The set of rules governing system communication constitutes the protocol. Some protocols also require status information to be exchanged to maintain coordination between various independently operating functions. For multiple processing systems, the manner in which the various elements are connected is referred to as coupling, of which there are two types. Loose coupling is generally characterized by disjoint primary memory, use of bit serial buses, and comparatively small interprocess communication bandwidth, where communication among distributed elements is accomplished by message passing. Tight coupling is generally characterized by shared, directly accessible, main memory, with comparatively high primary memory bandwidth and lower interprocessor communication latency. Loose coupling is typically easily modified and tightly coupled systems are typically difficult to modify.

Referring to Figure 2.2, the circles contained within the interconnection structure represent coupling mechanisms. Although to maintain flexibility all elements should have the potential for being connected, the actual connection/coupling scheme is determined by information flow and data rates of the application. For a loosely coupled configuration in which a bus is assumed to be connected to all elements in the system, the bus should be able to access all elements, with provisions to add more. Capability should also be provided to distribute simultaneously the information flow among several buses to accommodate increased loads.

Connection should also have provisions to allow communication on a local level with separate buses, so that the load on the global or system bus can be relieved and still maintain necessary communication among the system elements. Depending on the data rates, the local bus structures could also be loosely coupled.

The choice between loose and tight coupling is properly the subject of detailed analysis of the projected application requirements.

Task Intercommunication

In a multiple processing architecture, tasks will be distributed among various processing elements, as illustrated in Figure 2.2. The nature of G&C applications often makes these tasks interdependent, leading to the interaction of tasks in different processors, as well as within local processors. Certain common routines that are required by many tasks may not be replicated in all processors but, instead, be centralized in a globally accessible processor.

To facilitate flexibility, the location of resources necessary to execute a task should be transparent to the task. This implies the need for an address translation mechanism which coordinates the use of all resources needed by a task, including the processor and memory to which it is assigned. One means of maintaining this transparancy is to employ message packets for all intertask communications.

Executive Control Structure

An executive provides a control structure that is capable of directing all system resources to interact efficiently in accomplishing the overall G&C system function. The executive is responsible for functions such as the scheduling and dispatching of tasks, resource control, task intercommunication and control, and interrupt handling. In centralized systems, the executive control structure is primarily located in a single processor. Multiple processing system architectures will require the executive function to be distributed among the various processing elements.

One of the current issues regarding multiple processing architectures is that of dynamic or static load balancing. Dynamic load balancing is concerned with the dynamic assignment of tasks so that the workload tends to be equally distributed among the various processors. The control structure necessary to realize this type of architecture is extremely complex and is the subject of many investigative studies. In static load balancing, on the other hand, tasks are preassigned to specific processors at system generation time, so that the location of these tasks is always known. Tasks are reassigned to alternate processors only during failure reconfiguration, and even these assignments are deterministic. The control structure for an architecture with static loads is less sophisticated than for dynamic loads, because no dynamic workload distribution mechanism is necessary. Static load balancing is the more desirable approach because it more nearly matches the nature of the avionic processing problem and because it presents less of a run-time burden on the processing system.

As indicated previously, Figure 2.2 depicts a general structure for achieving system flexibility. Techniques for implementing this structure are subject to many architectural and application considerations. Architectural modularity, for example, must be accompanied by well-conceived interfaces for hardware, software, communications, and control in support of the application function being implemented.

2.2.2.2 ADAPTABILITY

During the life of a system there will be numerous changes in the nature of operations. Some of these changes may result from changes in the anticipated threat or from changes in tactics used to counter existing threats. Certain functions within the system may require updating or replacement by new functions. It follows that the processing system design must be able to accommodate or adapt to such changes in function with minimal hardware redesign or disruption of other ongoing functions in the system. The system must be able to accommodate software revision and the addition of hardware elements without change to the basic system design.

The priority of approaches to accommodate functional changes is, first, to add or modify subtasks within a task group and then to add new task groups. The rationale for initially evaluating the addition or modification of subtasks within a task group is to minimize additional hardware and, consequently, interconnection modifications. However, given a flexible interconnection structure, the addition of processor, memory, or I/O channels should also be accomplished with relative ease.

2. 2.3 FAULT TOLERANCE

The operation of modern military G&C systems and the aircraft itself depends so heavily on processing that failure of the processing function cannot be tolerated by the system. As a result, the system must be designed such that faults in processors can be masked from the system, and operation can proceed either with the full original capability or in an acceptable degraded mode.

The fault tolerance capability of the processing system will have significant impact on the probability of mission completion, and thus mission effectiveness. Implementation of a fault tolerance capability will, probably involve additional hardware, which is subject to failure. Thus the classical reliability parameter, Mean Time Between Failures (MTBF), may be degraded to the extent that additional components will increase the overall serial failure rate. However, these failures can be circumvented or masked on the system level by using fault tolerant techniques to increase system availability. The relevant maintainability parameters, Mean Time To Repair (MTTR), will probably be unaffected by fault tolerance. However, a prime requirement for fault tolerance is comprehensive automatic fault detection and isolation. This capability will significantly improve MTTR. It is uncertain how fault tolerance will affect availability parameters as presently measured. It depends on the decision either to fly with a masked fault or to repair the fault.

The ability of a system to recover from a fault with current technology is constrained by weight and cost factors. It is unrealistic to duplicate large and expensive sensors to provide better fault tolerance; therefore, it is unrealistic to expect to maintain

full performance in the face of any fault tolerance capability such that the MTBF of the computer exceeds the reliability of any of the sensors it may service by more than a factor of five or six. The advent of low cost (power, weight, etc.), microprocessor hardware may well produce a reversal in this area.

In cases where it is required that the system be able to handle a sensor failure without suffering a complete system failure. That is, the system must be capable of providing the maximum performance possible without the data normally provided by the failed sensor. Such "graceful degradation" can be provided for in the design of the architecture. Complete recovery from a processor fault can be attained through the proper use of redundant processing hardware and interface paths. Flexibility and growth can be built into the system so that increased or modified requirements can be met at minimum costs, or new technology can be easily incorporated to improve performance.

To date, little hardware has actually been produced to satisfy the requirements of a next generation Distributed Computer Controls System (DCCS) for G&C use. It should also be noted that the next generation missions, and environments will unquestionably be more demanding of load capacity than any system previously generated. The following discussion defines specific types of partitioning that can lead to successful systems alternatives for the next generation avionics G&C architectures.

2.2.3 GENERIC ARCHITECTURE TYPES

Three types of architectures were introduced earlier in this chapter and each will be discussed in this paragraph. To reiterate these architectural types were:

- Single Level Multiplex
- Hierarchical Multiplex
- n Level Multiplex

All of the architectures discussed in this chapter presume the use of at least one multiplex data bus such as MIL-STD-1553B. The basic difference between the three types of architectures is how many multiplex busses are employed and how are they interconnected.

There are a number of other elements which make up an architecture such as the multiplex bus control and the data transmission protocol.

For any multiplex bus system, there are two fundamental operations that must be performed. The first is the allocation of the bus, defined as the process of determining which processor is next entitled to assume control of the bus. There are a number of different methods of accomplishing this determination ranging from static to contention. These methods encompass no transfer of control to a priori transfer of control to demand transfer of control.

Let us characterize the types of transfer of control to be discussed herein terms of control transfer characteristics. There are four major types of interest. They are as follows:

- Static Bus Control
- Serial Polling Bus Control
- Round Robin Bus Control
- Contention Bus Control

Static Bus Control

With this method there is no transfer of control during system operation. A processor is designated as the bus master or bus controller and remains so during system operation. Since there is a possibility of failure and consequent loss of system control a number of schemes for back-up bus control have been devised. Probably the most prevalent is the employment of a "watchdog" timer where a designated processor watches the bus master to ensure that it is active and in fact controlling the bus. In the event that the bus master fails the "watch dog" processor will perform some minimum reconfiguration and assume bus control. A number of proposed architectures under investigation have been reported in current literature where this backup bus control function will be implemented via a microprocessor.

Serial Polling Bus Control

With serial polling, control of the bus offered to other processors in the system by the current bus controller. If the offer is not accepted, the current controller continues to offer the bus to other processors. Bus control is offered to processors until the offer is accepted. Upon the acceptance of the offer, the accepting processor conducts a message transaction and then deallocates the bus which becomes available for reallocation by the original controller. An alternative method is where the now current bus controller conducts the process of serial polling. The first case is defined as centralized serial polling and the latter decentralized.

Round Robin Bus Control

This method of bus control transfer is characterized by a logical ring structure where control passes apriori from one processor to the next in sequence until control passes back to the first processor in the sequence. This type of bus control transfer differs significantly from serial polling in that the current controller does not look for a processor willing to accept control. Instead the next processor in the ring must accept control whether it has need to transmit data or not.

Contention Bus Control

With contention bus control each processor that has a message to transmit competes for possession of the bus when it becomes free. Unlike serial polling buses, a contention bus represents the classical queuing situation where a priority discipline is in operation at the queue. Here, the queue is the conceptual combination of all messages waiting for bus service over all processors. Each message arrives at the conceptual queue bearing the priority of its processor. Whenever the bus becomes available, the message with the highest priority obtains immediate service.

The second operation is to conduct a transaction; i.e., transfer a message. To do so requires adherence to some form of protocol. Protocol is defined as a set or rules for the operation of a communications system. These rules are designed to solve operating problems in the following areas:

- o Framing — The determination of which bit groups constitute words (word length), and what groups of characters constitute messages (message length)
- o Error Control — The detection of errors by means of encoding checks, parity checks, message length checks, cyclic redundancy checks and so forth; the acceptance of correct messages and the request for retransmission of faulty messages
- o Sequence Control — The identification (may be sequence alone) of messages to avoid their duplication and loss, and the identification of messages that are retransmitted by the error control mechanism
- o Transparency — The transmission of data without causing the source or destination to modify the data.
- o Line Control — The determination, as in the case of a half-duplex or multipoint line, of which station(s) is (are) going to receive and transmit
- o Timeout Control — Solving the problem of what to do if message flow ceases beyond a specific time
- o Startup Control — The process of starting transmissions in a system

Protocols are divided into three categories, according to the framing technique used. They are as follows:

- o Character Oriented — A protocol that uses special characters, such as STX, to indicate the start of a message, and EOT to indicate the end of transmission
- o Byte Count Oriented — A protocol that uses a header, which includes a beginning special character followed by a count indicating the number of characters in the data portion of the message and some control information
- o Bit Oriented — A protocol that delineates which bits constitute messages by separating those messages with a special flag character

Table 1 reflects some of the features of selected transmission protocols.

Table 1. Current Transmission Protocols

Featured protocol	BISYNC	DDCMP	ADCCP	HDLC	SDLC	MIL-STD-1553
Sponsor	IBM	DEC	ANSI	ISO	IBM	DOD
Protocol Type	Character oriented	Byte count	Bit oriented	Bit oriented	Bit oriented	Bit oriented
Transmission Code	Half duplex	Full duplex	Full duplex	Full duplex	Full duplex	Half duplex
Transmission format	Serial	Serial/ parallel	Serial	Serial	Serial	Serial
Asynchronous operation	No	Yes	No	No	No	Yes
Error Detection	CRC-16	CRC-16	CRC-CCITT	CRC-CCITT	CRC-CCITT	Parity
Protocol overhead (single data message)	80 bits	80 bits	48 bits	48 bits	48 bits	40 bits or 80
Speed						

For the purpose of a specific application, one protocol may be more serviceable than another. For example, an increased processing overhead can be a penalty for the use of a busing system with a protocol not sufficiently matched to the needs of the application.

2.2.3.1 SINGLE LEVEL MULTIPLEX ARCHITECTURES

This architecture is typified by a single multiplex-bus which interconnects the total system. A major advantage to be obtained from this system over a non multiplex bus based system is the ability to logically replace one machine with another and thus achieve some measure of fault-tolerance. A means by which this ability can be achieved is via the principle of a bus master. With a bus master, the executive control function may be transferred from one machine to another in the same manner as one may pass a baton, thus transferring bus control from one machine to another. The software required to perform this function is more intricate than that needed for centralized systems and is correspondingly more expensive. Also, the loss of any particular processor may only lead to temporary loss of a function. This configuration permits a measure of graceful degradation for the system. With differing deterministic software load modules some level of loss could feasibly be tolerated with the loss in system capability being isolated to the loss of a subsystem.

One of the major drawbacks to this type of architecture that regardless of the type of bus control mechanism adopted, all bus controllers must be cognizant of every device in the system since they are all connected to the single multiplex data bus. The consequence of this is that the software must be modified to accomodate either the addition or loss of a device. There are methods of making these configuration changes relatively transparant (e.g., table driven software) to the software, but as the system grows even these methods become increasing complex.

Another limitation may well be the number of unique devices which can be accomodated by the multiplex addressing capability.

2.2.3.2 Hierarchical Multiplex Architectures

A hierarchical architecture is a structure of single level multiplex architectures cascaded from a primary level to the most secondary (subordinate) level in the hierarchy. The overall complexity of this system can be a direct function of the number of layers of control required and the software associated with control. One means of controlling the diversity and complexity of the software would be to treat each level as a subsystem and restrict communication to/from this subsystem to integrated or processed data. This mechanism would eliminate levels of control since each processing element communicating with this "subsystem" would only be cognizant of integrated data packets and not the sources of raw data. This configuration is expected to receive widespread examination in the future because of its flexibility. For multifaceted or greatly extended and complex systems, this configuration offers the system designer an opportunity to control this complexity by a "natural" partitioning of the system. In Figure 2.3 it can be seen that the architecture is essentially a cascading of federated configurations.

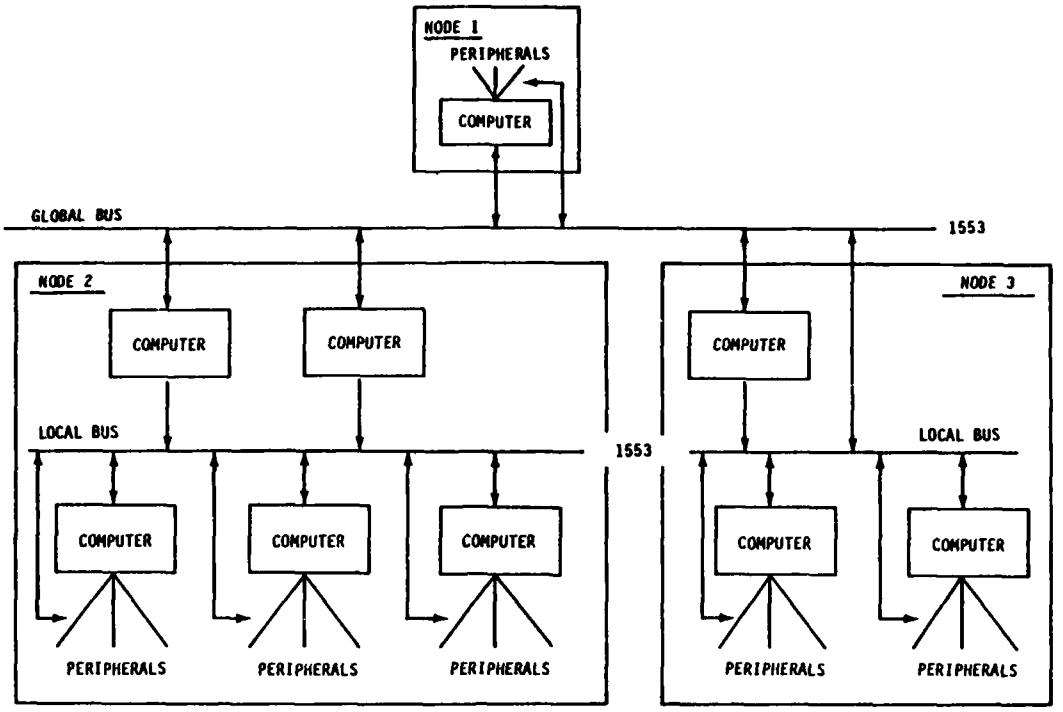


FIGURE 2.3 CONCEPTUAL HIERARCHICAL STRUCTURE

Distributed processing systems are applicable to realtime dedicated functions where the problem has a natural division of subfunctions that can be easily decomposed into separate tasks and assigned to multiple processing elements. G&C systems generally exhibit these characteristics since the avionics can be partitioned into functional regions such as navigation, communications, IFF, flight control, and mission display/information management where each region is typically composed of multiple elements. In general, the major considerations in partitioning system processing functions would be the maximization of system availability and the minimization of data transfers between regions. Thus, system processing functions that tend to support one particular subsystem more heavily than the others would be included within that subsystem's region.

A method for interconnecting partitioned functional regions of an avionic system is by means of a hierarchical structure is conceptually characterized in Figure 2.3. As with any multiple element configuration, the various design issues that need to be addressed include functional allocation, processor selection, processor interconnection, data base organization, system control, reliability, expandibility, and software characteristics. These issues are application dependent and their resolution is subject to detailed analysis. The intent here is to discuss general characteristics and alternatives relative to a hierarchical avionic system configuration on a qualitative level.

A design concept for a hierarchical structure is to provide the capability for subsystem peripheral data to be accesible by many processors either directly or through another processor despite one or more failed processors. This is accomplished through the use of a set of peripheral buses. These multilevel peripheral buses allow flexibility in the distribution of the processing load and in the addition or modification of peripherals.

Since the hierarchical structure permits distributing data flow among various local buses, activity on the global or system bus is minimized, thus allowing for future expandability or growth. This enables a two-dimensional growth potential, i.e., in the horizontal and vertical directions. Horizontal expansion refers to growth along a single level multiplex bus. Vertical expansion refers to growth in terms of additional bus levels. From an implementation point of view, the number of secondary or local bus levels should be minimized to reduce complexity.

Referring to Figure 2.3, it can be readily seen that a multitude of interconnect options are available. The choices for interconnecting computers and peripherals are dependent on the application and the degree of fault tolerance or system availability required and the associated reconfiguration scheme selected. Several possible interconnect alternatives are suggested as nodes 1, 2, and 3 in Figure 2.3. It may be

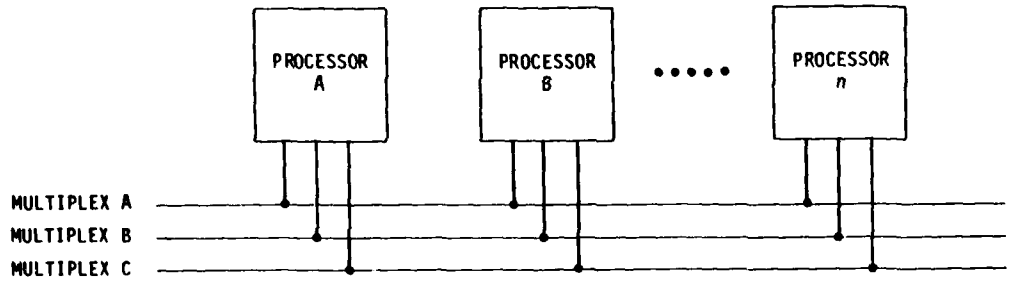
noted that the peripherals, which represent sensors/subsystems, can be connected either directly by means of an I/O channel, or by means of some bus configuration. However, from a commonality point of view, it is highly desirable that all buses be of the same type. The scheme shown for node 1 indicates data distribution to the global bus by means of a computer. However, should the computer fail, a secondary path is provided. It should be expected that an increased bus loading would result from use of the secondary path and that the required processing would be assigned to another processor. An alternate scheme is depicted as node 2 where another processor is available to perform the preprocessing function thereby eliminating the need for a bypass path to the global bus. Another variation contained in node 2 is that two levels of computing are provided and that all computers on the lowest level can access all data furnished by the peripherals. Node 3 is similar to Node 2 except that a bypass path is provided in lieu of a computer.

The processors envisaged for implementing future multiple processing avionic systems are either an advanced, light-weight, higher performance version of a standard instruction set architecture computer, possibly resulting from a high performance logic development program, or an advanced standard 16-bit microcomputer. The computer would be expandable in terms of memory, hardware floating point capability, and I/O processing. However, it is highly desirable that the processor architecture be identical, and any program executable on a lower capability machine would also be executable on a higher capability machine.

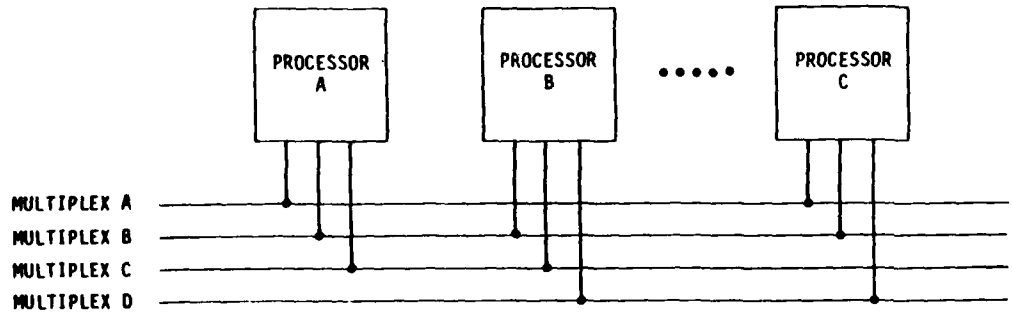
2.2.3.3 n LEVEL MULTIPLEX ARCHITECTURES

This architecture is primarily a variation in interconnect structure and can consist of a conglomerate of imbedded single level multiplex or hierarchical architectures. This "architecture" is included in this discussion because of evidence that it is being considered as a candidate architecture by a number of G&C system designers and the rapid advances in microprocessor hardware may well make architectures such as these attainable. This particular architecture seems to be typified by the processors with bus control ability connected to all of the n multiplex busses or in the case where it is not practical to connect to all they are connected to a large subset so there are multiple processors connected to each of the multiplex buses. This is illustrated in Case A and B below.

Case A:



Case B:



Similarly, each subsystem or remote terminal is connected to more than one multiplex data bus. The design logic supporting this architecture appears to be maximum redundancy. Although first examination appears to reveal maximization of the most reliable element in a multiplex system (e.g., the multiplex) in actuality it attempts to maximize the interface between the processing intelligence and the multiplex. Ongoing studies in the United States and other member nations are defining VLSI implementations of this interface packaged on a 1/2 ATR card which makes these multiple interfaces very affordable.

The discussions in the preceding two paragraphs apply to this architecture because the system designer can actually imbed hierarchical or single level designs into this architecture, depending upon the interconnect structure and the bus control mechanism implemented.

Figure 2.4 illustrates a classic triple redundant multiplex architecture for G&C on a tactical airframe. Examination of this figure reveals numerous opportunities for implementation of a n Level Multiplex Architecture depending upon the bus control structure selected. For example consider the display management subsystem on Multiplex A being controlled by any one of the G&C computers on Multiplex A with sensor data (both raw and processed) being provided to the display subsystem. This is actually an example of a hierarchical interconnect even though the display subsystem elements are not distributed on a multiplex bus. Next, consider the flight system on Multiplex B and C being controlled by the other two G&C processors with voting being accomplished through DMA. This is actually two single level multiplex applications.

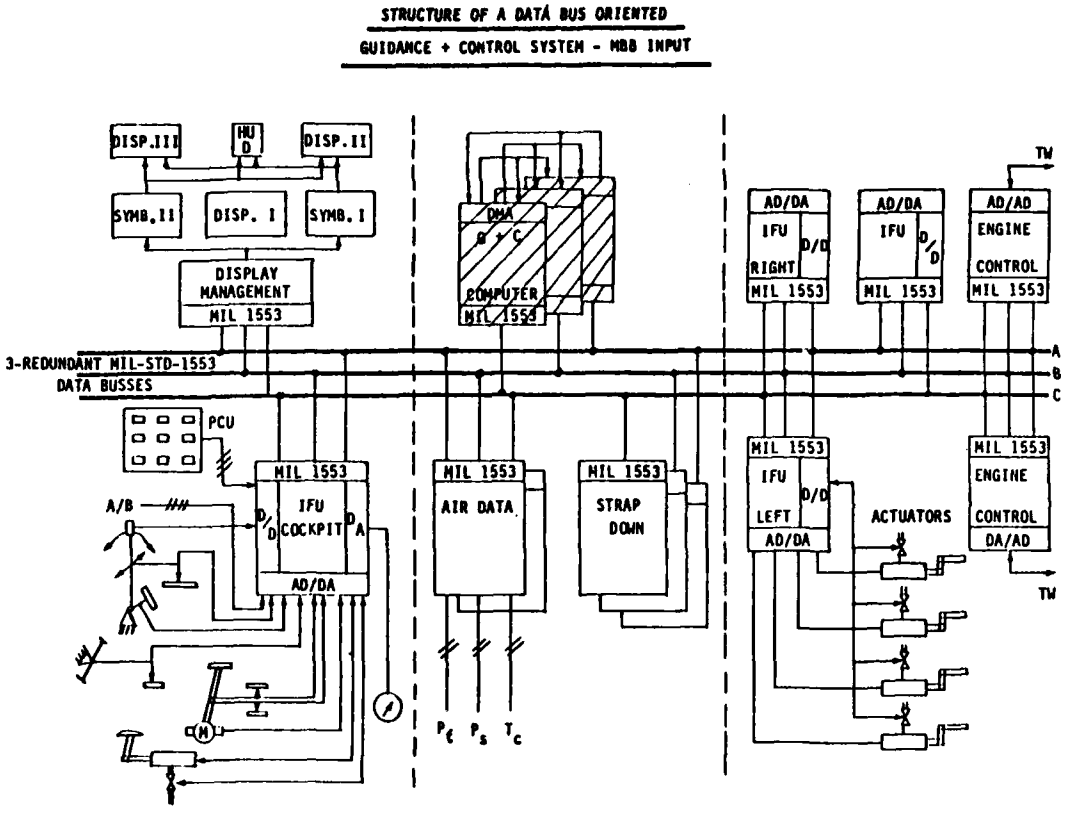


FIGURE 2.4

Variations of this structure can possibly provide the maximum amount of system availability and reliability at a minimum of cost in terms of size, weight, cooling and so forth since the redundancy is effected through the multiplex interface hardware. Current activities in this area are packaging the multiplex interface into VLSI devices which may well allow implementation of such architectures.

Regardless of the advances in microprocessor technology the system control and necessary software to implement this type of architecture will necessarily be extremely complex.

2.3.0 SAMPLE ARCHITECTURES

The advent of the microprocessor, along with the development of semiconductor memories, has created a revolution in aircraft/missile G&C systems design practices. It has become apparent that there are limitless configurations which can be assembled through use of distributed architectures. The sample of G&C systems designs presented in this section exhibit the general architectures introduced in the preceding section. Some of these architectures are in existing air frames and some are still on paper but all are candidates for update with microprocessor hardware.

The emphasis in G&C system design has shifted from the use of a single multi-function central computer to that of a distributed network of unit processors, whereas single function or a small set of similar or related functions are embedded in a processor. This illustrates what was stated in a previous section, the major impact of microprocessors upon G&C system design is the use of microprocessors to implement distributed networks.

2.3.1 COMMERCIAL AIRFRAMES

Microprocessors are rapidly finding a home in commercial airframes, particulary in the area of airframe related avionics. Most of these systems are disjoint and little has been done relative to integrating them via a multiplex bus for shared data. A recent study completed for the USAF revealed over 120 processors on a new generation airliner. What is apparant from this particular study is the need for standardization to avoid rampant integration and O&M costs due to a proliferation of hardware of non-standard types. Probably more critical is the proliferation of increasingly expensive software. The need for standardization is being recognized however and efforts will soon be underway to examine integration concepts similar to those being implemented in modern military avionics systems.

One such effort is being undertaken by NASA Langley's AIRLAB. This effort is developing a hot bench laboratory to investigate integration via multiplex data bus based architectures on commercial airframes. Examples of the types of architectures to be investigated in this facility are illustrated in Figure 2.5.1 and 2.5.2.

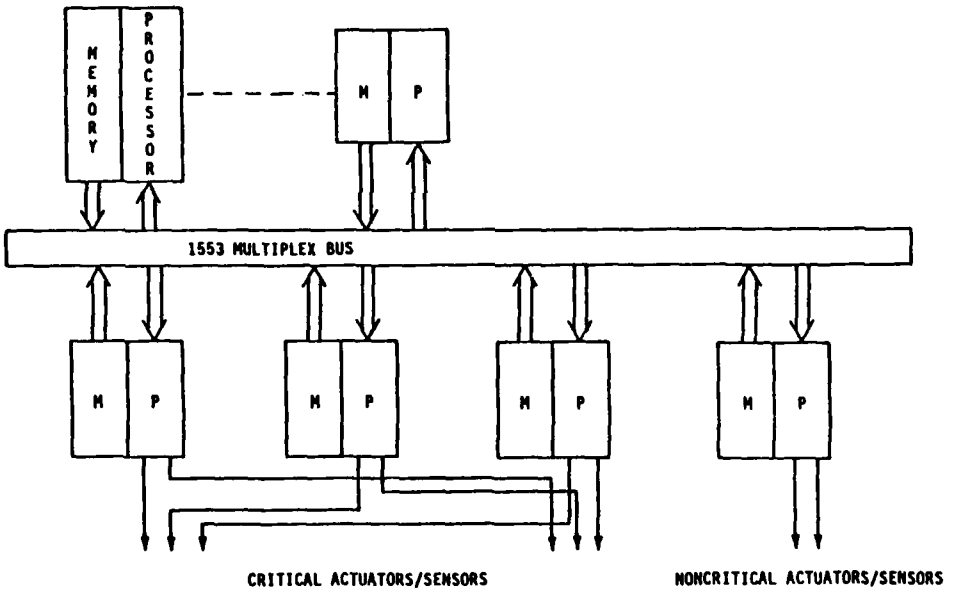


FIGURE 2.5.1 FAULT TOLERANCE IMPLEMENTED VIA SOFTWARE

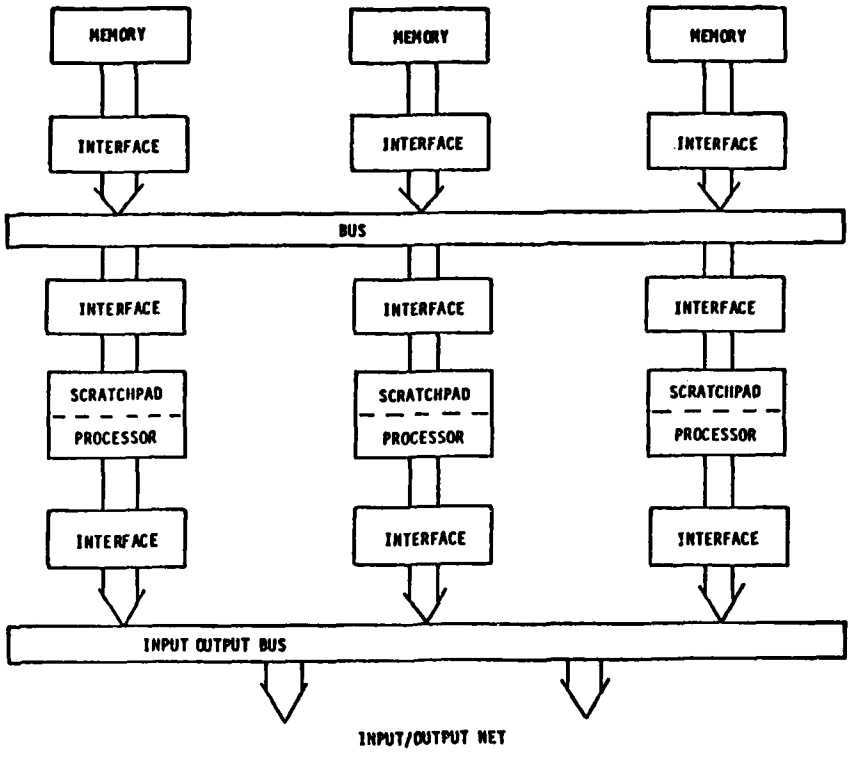


FIGURE 2.5.2 FAULT TOLERANCE IMPLEMENTED VIA HARDWARE

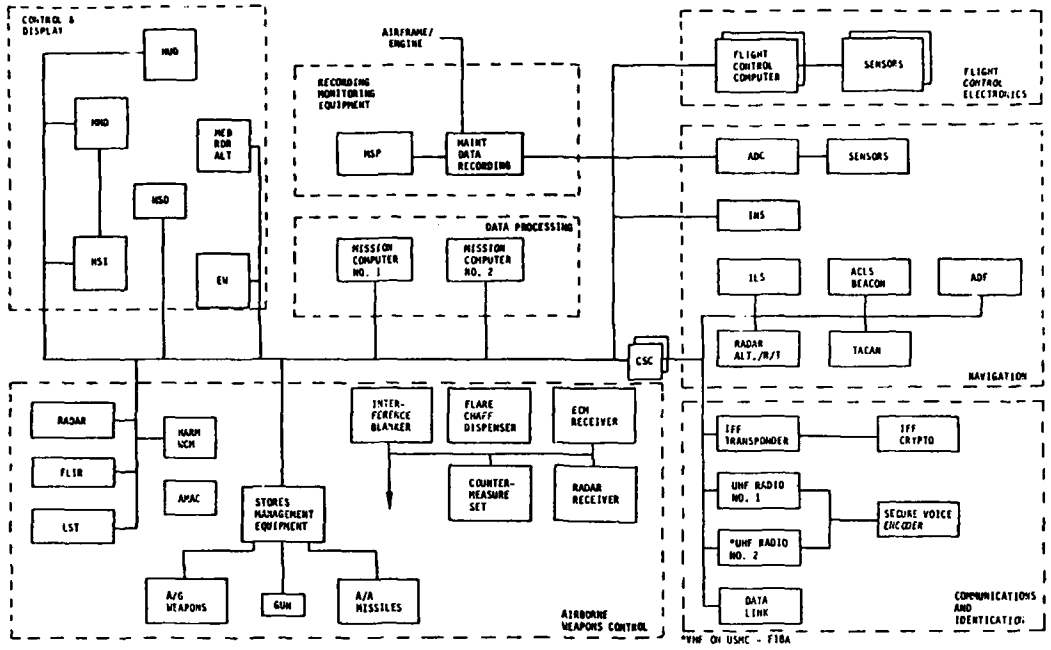
2.3.2 TACTICAL AIRFRAMES

Tactical aircraft, such as the F-4, the A-7, the F-5, etc., impose a tremendous cockpit load on the pilot.

Probably for this reason we have seen the most work in advanced G&C architectures in the tactical fighter aircraft area. Many different tactical airframes employ architectures using multiplex data buses as the data transfer medium and a variety of bus control schemes. Many of these architectures are candidates for use of microprocessors and investigation of the use of microprocessors is underway at the present time. One good example of this investigation is the VLSI MIL-STD-1750 program on the USAF F16 MSIP program that is ongoing.

The G&C architecture for the USN F18 tactical fighter aircraft is exhibited in figure 2.6. This particular architecture is an example of a federated single-level multiplex bus architecture. Some characteristics of a hierarchical structure is exhibited in the CSC area. Figure 2.7 exhibits a speculative future architecture for this same aircraft, employing a hierarchical structure probably utilizing much state of the art microprocessor hardware primarily in the areas of simulataneous bus control, processors and intelligent sensors/subsystems.

FIGURE 2.6 F-18 AVIONICS SYSTEM



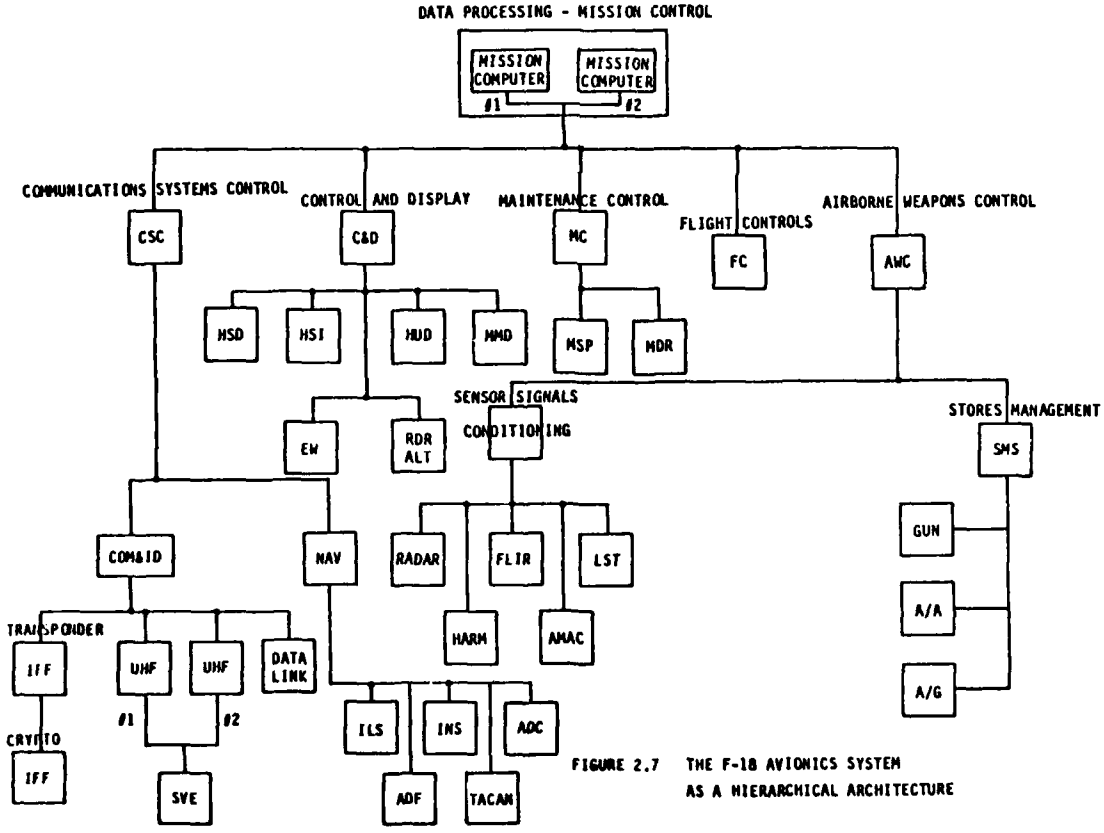


FIGURE 2.7 THE F-18 AVIONICS SYSTEM AS A HIERARCHICAL ARCHITECTURE

2.3.3 STRATEGIC AIRFRAMES

An architecture employed on a modern day strategic airframe is exhibited in figure 2.8. This is basically a hierarchical structure comprised of two single level multiplex architectures interconnected by the processors A & B. Each bus is controlled by a separate processor (Processor A controls the Navigation and Weapon Delivery bus and Processor B controls the Controls and Display bus). The processors that handle bus control also handle and manage the transfer of data between the two multiplex buses.

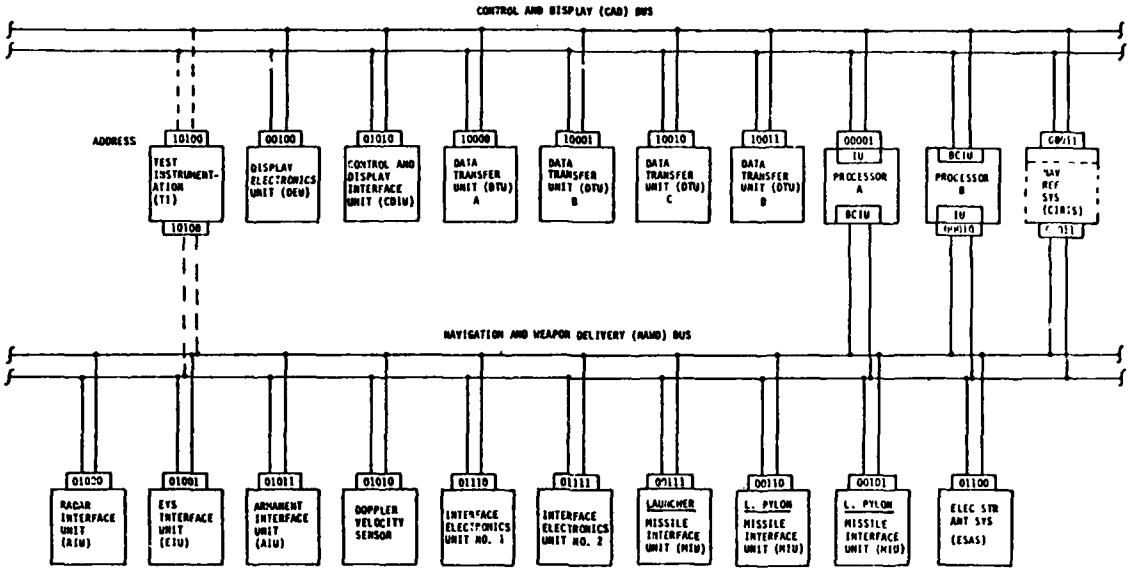


FIGURE 2.8 OAS MULTIPLEX SYSTEM ARCHITECTURE

In the event of failure of one processor the other will assume the bus control function for the failed processor, providing some degree of fail safe or fault tolerant operation. The remote terminals in this architecture (identified by five bit addresses) are all candidates for microprocessor based intelligent remote terminals that provide processing capability at the weapon sensor actuator to provide some degree of autonomy (isolation) and to unburden the mission processor.

2.3.4 TRANSPORT AIRFRAMES

Review of available literature has revealed that there are no operational military transport airframes employing architectures with multiplex data buses. The USAF is proceeding with the design of a new transport aircraft in which the Guidance and Control architecture will include a MIL-STD-1553B data bus as the common data transfer medium. Further the USAF request for proposal clearly mandated that common microprocessors and a common Higher Order language would be specified in the design. The contract award was announced in August, 1981 and there is virtually no data to report at this time. Speculation on the part of the authors, however, indicates that the DOD MIL-Standards will be employed, including MIL-STD-1553B, MIL-STD-1589B and MIL-STD-1750A and standard (common) microprocessors. There is a good possibility that information on the proposed architecture will be available in the near future.

There is also some activity ongoing to install a MIL-STD-1553B multiplex data bus on the USAF KC135 tanker but this activity is still in competition; consequently there is not any data to report relative to this activity.

At the present time there is an effort funded by the Aeronautical Systems Division of the USAF to construct a hot bench laboratory for KC135 architectural evaluation and study. This proposed laboratory is built around a MIL-STD-1553B multiplex data bus, using MIL-STD-1750A processors programmed in MIL-STD-1589B JOVIAL J/73. The laboratory will also include a suite of real and simulated sensors. Even though this is a laboratory construction effort it may well indicate the direction of future architecture definitions for the KC135 airframe

2.3.5 MISSILE AIRFRAMES

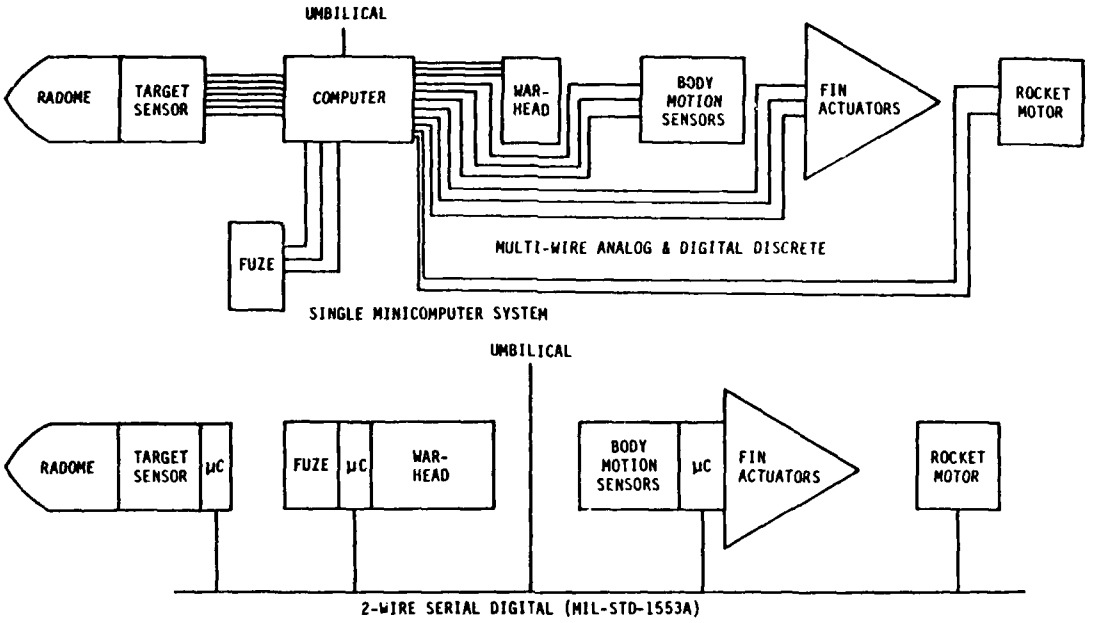
Like aircraft designers, missile designers had always wanted to inject digital logic into their designs. But missile designers had to wait until the transistor had been reduced further in weight and power consumption. With the advent of integrated circuits, custom designed controllers and programmed logic circuitry became a practicality. With the microprocessor, the missile designers are coming into their own. And like the aircraft, the missiles had to settle for centralized computer structures at first, because the available machines all had limited throughput. Typical of these early systems is the Navy's Hi-Speed Anti-Radiation Missile (HARM) which includes the Texas Instruments TI-2501 microprocessor.

Despite the advances in IC technology, there are few examples of true distributed computer systems in the missile world. Nevertheless, the shortcomings of single-computer systems are producing a strong motivation for distributed systems. The principal shortcomings can be identified as:

- o Loss or subsystem autonomy
- o Expense of software
- o Limited opportunity for growth

In Figure 2.9 we see a comparison of single-computer and distributed computer architectures for missile avionics systems. Note the simplicity of the distributed system. The single-bus federated system is typical of most contemporary designs and trades speed for simplicity. The USAF is considering just such a federated approach as a solution to the navigation problem for the Air Launched Cruise Missile (ALCM). It is called Digital Integrating Subsystem (DIS) and utilizes several microprocessors as unit processors.

DISTRIBUTED MICROCOMPUTER (μC) SYSTEMS SIMPLIFY SUBSYSTEM DESIGN, MANUFACTURE, INTERFACE, TEST & UPDATES



DISTRIBUTED MICROCOMPUTER (μC) SYSTEM

FIGURE 2.9

Both the USAF and USN are contemplating fully distributed systems designs for missiles. Navy has invested six years and several millions of dollars in its Modular Digital Guided Missile System (MDGMS). The US Air Force has done likewise on its Digital Guided Weapons Technology (DGWT) program. The most recent development in this program is a fully hierarchical system.

2.3.6 SURVEILLANCE AIRFRAMES

Figure 2.10 illustrates a candidate S-3A architecture. It may be classified as a homogeneous hierarchical system, since it uses standard computers interconnected via a hierarchy of buses. The system is derived from the processing requirements of the S-3A aircraft. The primary requirements are its ability to meet the performance requirements, availability in the event of processor failure, expandability to accommodate new or changing requirements, and compatibility with hardware and software standards.

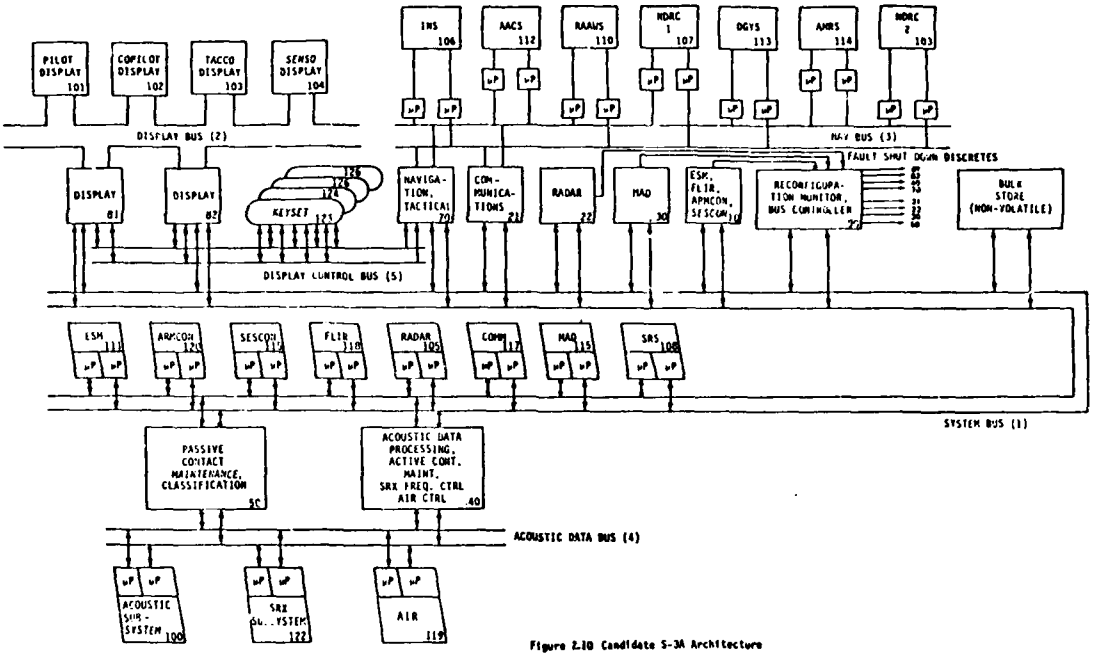


Figure 2.10 Candidate S-3A Architecture

Toward these objectives, the system is oriented around a redundant set of buses used for subsystem-to-subsystem communication, with separate secondary buses used for computer/peripheral communications. Peripheral interfaces with a secondary bus are implemented using identical redundant microprocessors, which, in a platform application, are projected to be implemented on a single board. These microcomputers are redundant, and are controlled by the minicomputers attached to their particular secondary buses. Thus, the interfaces can be made standard and identical, enabling easier reconfiguration and expansion.

Availability is provided by redundant hardware, redundant data paths, and reconfiguration software stored in the minicomputers and in a bulk storage medium accessible to all the processors. Additionally, one computer is dedicated to the test/monitor function and can be used as a spare in case of a processor failure.

The candidate architecture represents a system in which cost, flexibility, and availability are considered more important than weight and processor speed. Future requirements may be accommodated through the use of additional computers rather than through the use of excess processing capability within the existing computers. This approach reduces the cost of programming, since interaction between modules, especially for the purpose of increasing real-time speed of response, is minimized.

2.3.7 V/STOL AIRFRAMES

Figure 2.11 depicts a conceptual hierarchical configuration for a V/STOL "Type A" data processing system. This system would be outfitted with either ASW (Antisubmarine Warfare) or AEW (Airborne Early Warning) electronic systems. The assumptions under which this system is configured are described in the following paragraphs.

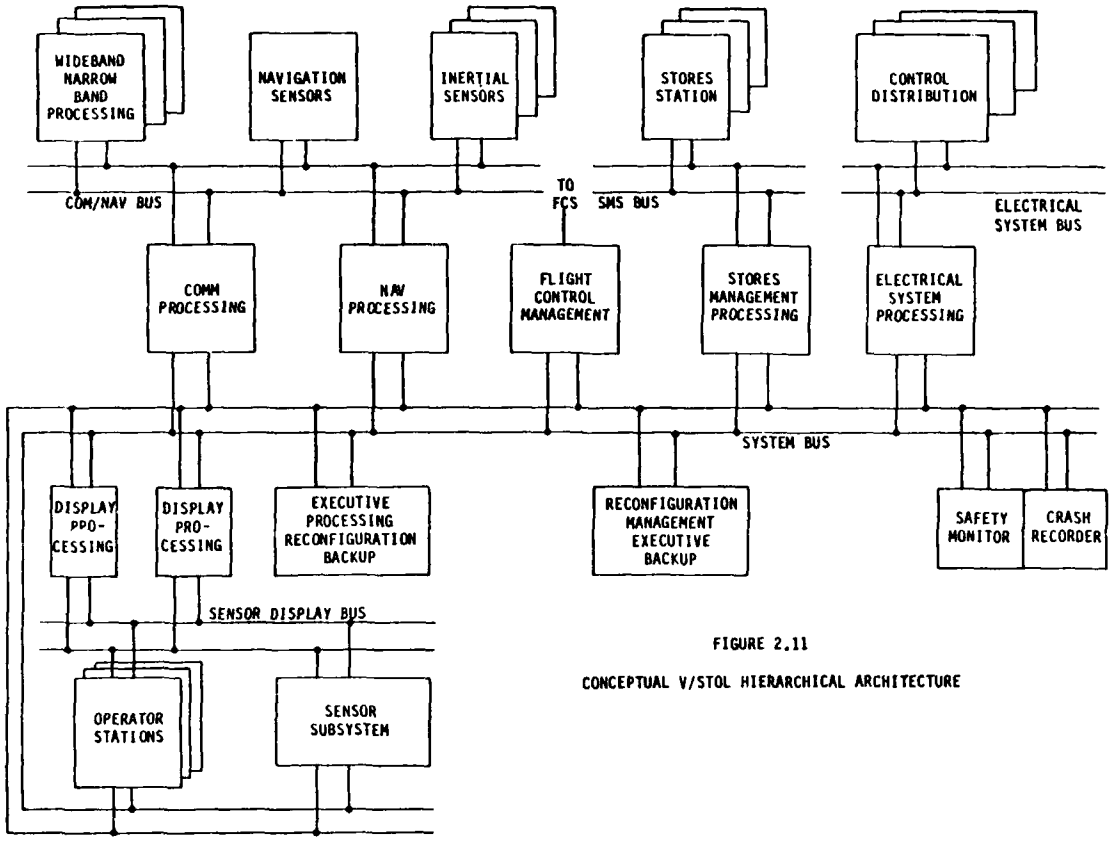


FIGURE 2.11

CONCEPTUAL V/STOL HIERARCHICAL ARCHITECTURE

The processors shown are standard units, identical in architecture and I/O (with the exception of the display processors' video interface and the flight control interface). Any program executing in one processor can execute in another (although not, perhaps, at the same speed).

The subsystem processors are assumed to have enough hardware and software self-contained test to know when they have a fault, report this condition to the system bus (perhaps merely by not responding to a status request), and to isolate themselves from the system (perhaps by removing power).

The Communications/Navigation bus, the Stores Management bus, the Electrical System Management bus and the System bus are all identical, serial multiplex buses of the type envisioned by MIL-STD-1553. However, the speed requirements will be on the order of 10 M bits/sec, which is well within the state-of-the-art for either fiberoptic or coaxial transmission lines. A higher bus speed (50 M b/sec) would be desirable to insure the maximum fault tolerance capability, since transfer of recovery programs could be accomplished with little interruption of normal processing functions.

The major design concept in this type of configuration is the ability for subsystem peripheral data to be accessible to many processors, either in normal operation or in the presence of one or more failed processors. This is accomplished through the use of a set of peripheral buses. These second level peripheral buses allow flexibility in the distribution of the processing load and in the addition or modification of peripherals.

2.3.8 Helicopter Airframes

Microprocessors are also finding their way into helicopter airframes. Figure 1.12 illustrates a typical configuration using the Sperry Flight Systems SDP 175 Microprocessor. Typical applications include fire control, stores management, avionic data processing, Guidance and Control and control of a multiplex data bus.

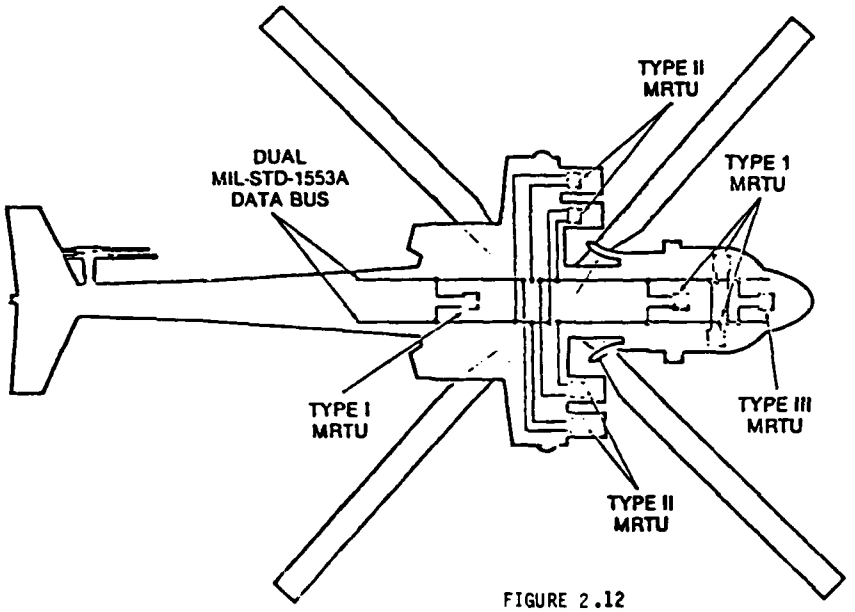


FIGURE 2.12

2.4.0 ENGINEERING JUDGMENT AND DISCUSSION

In the previous paragraphs of this chapter some background was provided on the impact of microprocessors upon guidance and control systems for future airframes. Additionally possible architectures were categorized into three generic types - single level, hierarchical, and n-level. Examples of architectures for various classes of aircraft were presented.

In this paragraph a high level trade off will be made of the three types of architectures, using a tactical fighter aircraft as the proposed vehicle.

2.4.1 ASSESSMENT CRITERIA

The final selection of a guidance and control architecture for a tactical fighter aircraft would be subject to years of research and study by highly qualified engineering personnel. Any attempt to make such a selection in a report such as this would be presumptious. Consequently, the discussion will be limited to a relative discussion of the three types of candidate architectures. A set of assessment criteria were selected based upon those facets of an architecture which are important in the specification of an architecture.

The assessment criteria are:

ADAPTABILITY The ability of an architecture to be adapted to fulfill additional or different mission requirements without significant hardware redesign or disruption of other functions within a system.

BROADCAST CAPABILITY The ability to transmit a single message or data packet to multiple users with a minimum of system resources (including both hardware and software).

COMMONALITY The ability to maximize use of common elements throughout the system to minimize costs associated with a system which includes acquisition and operational and maintenance costs.

FAILURE EFFECT The overall effect or impact upon a system due to the failure of an element involved in information transfer or in information switching logic.

FAILURE RECONFIGURATION The ability of the system to reconfigure around a fault or failure. This is often referred to as fault tolerance.

FLEXIBILITY The capability of a system to be adapted to varying mission requirements without requiring additional software or hardware elements.

LATENCY The time necessary to transmit data from a source to a destination. In this context the definition is narrowed to mean the amount of time imposed by the system architecture upon the transfer time.

LIFE CYCLE COST The total costs associated with the system, including development, installation, operation, logistics support and maintenance. The costs considered in this report are primarily those imposed by the architecture being considered.

LOGICAL COMPLEXITY The complexity of the hardware and software logic necessary to support information flow to achieve communication throughout the system.

OVERHEAD The amount of time and processing resources devoted to system (bus) control and data transmission.

PERFORMANCE Information processing capacity.

PLACE MODULARITY The degree to which the addition of system processing nodes is restricted.

POWER CONSUMPTION The amount of power consumed by the system elements, including both processors and subsystems.

REDUNDANCY This parameter refers to the ability of the system to maintain normal degraded modes of operation in support of full or limited mission requirements in the event of the loss of system elements.

RELIABILITY The ability of a G&C system to continue to operate at a planned frequency and to produce the desired necessary data at a given location from a set of inputs.

RISK The degree of risk associated with the implementation of a system configured under the architecture being considered.

SECURITY The ability to restrict access to or generation of information by unauthorized system elements.

SIZE The amount of space necessary to install a particular implementation on an airframe.

SOFTWARE COMPLEXITY The degree of complexity made necessary by the implementation of a particular architecture.

STANDARDIZATION The capability to incorporate standard system elements into a system configuration. This ability can be construed negatively by determining how standards inhibit growth or incorporation of new technology due to constraints imposed by the architecture.

SURVIVABILITY The ability of maintaining mission and flight critical capabilities after suffering limited damage to some system elements.

TESTABILITY The ability to introduce data into a system and measure and observe desired system behavior, data processing and data transmission.

TRANSPORTABILITY The degree to which a system could be applied to another platform; conversely, the lack of applicability due to constraints imposed by the architecture.

WEIGHT Physical weight characteristic of the elements comprising the system.

Table 2 summarizes a high level trade of these assessment criteria as applied to the three candidate architectures presented in this section of this report.

TABLE 2

QUALITATIVE ASSESSMENT OF ARCHITECTURE ALTERNATIVES

ASSESSMENT FACTOR	GENERIC G & C ARCHITECTURE TYPE		
	SINGLE LEVEL	n-LEVEL	HIERARCHICAL
ADAPTABILITY	GOOD - due to functional partitioning and typical localized bus control.	FAIR - probably the worst of the three candidates because of the complexity of bus control to implement this architecture.	GOOD - probably the best of the three architectures because of the degree of localization of functions.
BROADCAST CAPABILITY	VERY GOOD - because each element is able to listen to a common bus, however, in most implementations of this architecture broadcast is prevented due to command/response protocol.	FAIR - the ability to broadcast in this architecture is determined by the information transfer paths available in the implementation. A message could be sent to every element in the system with very complex information transfer algorithms.	POOR - within a given level of hierarchy a message could easily be broadcast depending upon the information transfer protocol. The transfer between hierarchical levels would require complex bus control algorithms.
COMMONALITY	GOOD - bus control interface requirements are minimal and the remainder of the system elements could be common with respect to the bus, the subsystems being supported would dictate different but localized interfaces.	FAIR - varying degrees of bus control/response circuitry made necessary by this architecture would require a number of different elements, subsystem interface would be the same as for federated.	FAIR - due to the varying degrees of bus control bus response necessary in the various system elements. Most elements which interfaced to subsystems could be common but would differ from the bus control/response elements and the subsystem interface would have the same requirements as the other two candidates.
FAILURE EFFECT	POOR - in this architecture failures at the lowest level are known at the highest level resulting in increased control complexity.	GOOD - failures can readily be accommodated due to redundant information paths made possible through relatively complex control.	GOOD - at lowest level since failure is isolated at the lowest level. At the intra-level level the effect is dependent upon the redundancy at the intra-level element.
FAILURE REDUNDANCY	POOR - due to typical single point failures in implementations of this architecture.	GOOD - due to multiple levels of redundancy through data paths.	FAIR - possible single failure points in intra-level transfer elements.
FLEXIBILITY	Similar to adaptability.	Similar to adaptability.	Similar to adaptibility.
LATENCY	GOOD - direct information flow is provided between elements providing that high levels of asynchronous activity do not prevent data transfer.	VERY GOOD - providing data flow is maximized through utilization of efficient data paths. VERY POOR - if data must "sub" a path through the redundant paths.	POOR - if information flow is not optimized at hierarchical levels. GOOD - at individual hierarchical levels.
LIFE CYCLE COST	GOOD - probably the lowest because of the lack of complexity necessary in the system elements.	POOR - due to the degree of complexity necessary in the system elements.	FAIR - medium level of complexity dependent upon the degree of hierachical decomposition.
LOGICAL COMPLEXITY	GOOD - data paths are predetermined but alternate paths are not possible.	GOOD - for predetermined data paths. POOR - for alternate paths even though paths are available determination is complex.	FAIR - data paths can be predetermined and require the most complexity. Alternative paths are available with no significant penalty.
OVERHEAD	GOOD - data transmission control can essentially be implemented in tables.	FAIR - highly dependent upon the bus control mechanism and protocol.	FAIR TO POOR - Depending upon the level of upward reporting dictated by the implementation.
PERFORMANCE	GOOD - due to functional partitioning.	GOOD - due to functional partitioning.	VERY GOOD - due to functional partitioning and functional isolation.
PLACE MODULARITY	GOOD - provided adequate bandwidth is available.	VERY GOOD - sufficient bandwidth should be available on a number of busses.	VERY GOOD - expansion is available either horizontally or vertically.
POWER CONSUMPTION	FAIR - probably the highest due to the size of processors however the processor size is technology dependent.	FAIR - same as federated.	GOOD - probably the lowest due to the smaller required size of processors, however the size is technology dependent.
REDUNDANCY	FAIR - probably the poorest due to lack of redundant elements.	VERY GOOD - many alternate paths are available.	FAIR - proliferation of elements becomes necessary at intrabus levels.
RELIABILITY	FAIR - typical of this architecture is minimum redundancy and reconfiguration capability but data integrity and operational frequency good through proof.	GOOD - highest level of redundancy and reconfiguration capability, data and system integrity are subject to study.	GOOD - if implemented as a series of single level architectures, however overall frequency and data transmission frequency is subject to much debate.
RISK	VERY GOOD - proven technology.	POOR - control system complexity is not yet defined.	GOOD - each bus level can use proven federated technology.
SECURITY	POOR - in theory all data on a bus is available to entire system.	GOOD - data can be isolated to those elements on common busses.	VERY GOOD - data can be isolated to an hierarchical level.

TABLE 2 (Cont.)

QUALITATIVE ASSESSMENT OF ARCHITECTURE ALTERNATIVES

ASSESSMENT FACTOR	GENERIC G & C ARCHITECTURE TYPE		
	SINGLE LEVEL	n-LEVEL	HIERARCHICAL
SIZE	VERY GOOD - fewest elements but technology dependent.	FAIR - much redundant hardware but a function of technology.	FAIR - the most hardware but technology dependent.
SOFTWARE COMPLEXITY	FAIR - structure is fairly rigid but proven.	POOR - requires extremely sophisticated control mechanisms.	GOOD - complexity is limited to intrabus controllers which is a proven commercial technology.
STANDARDIZATION	VERY GOOD - proven by MIL-STD-1750, MIL-STD-1553 & MIL-STD-1589.	FAIR - probably requires new hardware (non standard).	GOOD - same as federated.
SURVIVABILITY	FAIR - least amount of functional isolation.	VERY GOOD - many alternate paths for data to allow maximum reconfiguration.	GOOD - due to functional isolation.
TESTABILITY	GOOD - assuming no asynchronous bus transactions all probabilities and cases should be able to be tested.	FAIR - the number of test cases could be an insurmountable number and the amount of test hardware.	FAIR - provided that the number of intrabus transfers is limited and predictable. The driving software and hardware is complex.
TRANSPORTABILITY	VERY GOOD - proven on many airframes.	Needs analysis.	GOOD - maximum degree of repartitioning capability.
WEIGHT	VERY GOOD - same considerations as size.	FAIR - same considerations as size.	FAIR - same considerations as size.

CHAPTER 3

MICROPROCESSOR AND DIGITAL SYSTEMS TERMINOLOGY AND NOMENCLATURE

by

Dr Thomas B CUNNINGHAM
Honeywell, Inc
MN17-2375
2600 Ridgway Parkway
P O Box 312
Minneapolis, Minnesota 55440

U S A

3. MICROPROCESSOR AND DIGITAL SYSTEMS TERMINOLOGY AND NOMENCLATURE

3.1 INTRODUCTION

The development of microprocessors for aerospace applications has been revolutionary. We are witnessing an equally revolutionary development and proliferation of terminology. Much nomenclature originated from computer scientists and developers. However, the aerospace user has in many cases, narrowed a wider interpretation of a given term for the avionics community or invented new definitions altogether.

The foregoing list is neither comprehensive nor in some cases precise to origins. The list contains commonly used and newly emerging terms along with the definition (in some cases more than one) generally accepted by international aerospace users.

3.2 TERMINOLOGY AND NOMENCLATURE

A/D conversion - Analog to digital conversion. The method used to provide a digital computer with information about an analog signal. It normally uses a voltage comparison to quantize an analog signal at a fixed rate of sampling. The assumption made is that the analog signal is a "well behaved" function between sampling points.

A/D converter - A device, usually an integrated circuit, which performs the analog to digital conversion function.

ALU - Acronym for the arithmetic logic unit.

AND - A logic operation that performs conjunction. (A AND B) = 1 only if both A=1 and B=1. Symbolic representations A•B and AB are commonly used.

ASCII - An acronym for American Standard Code for Information Interchange. A seven bit code that can represent an upper or lower case letter, numeral, or certain graphic symbols along with non-printable characters called control characters. (e.g. 0100001 = A, 0100010 = B, etc.).

Access time - The device dependent time between enabling a memory circuit and the data appearing on the output.

Accumulator - A register in a central processing unit which accumulates the output of the arithmetic logic unit.

Active High Signal - Denotes a signal that is active or true when it is in the high or logical 1 state.

Active Low Signal - Denotes a signal that is active or true when it is in the low or logical 0 state. An active low signal is usually denoted by a bar over the signal name.

Address - The means by which a computer locates a piece of information or data, or alternately the location in which a piece of information may be found.

Address Bus - The wires or conducting paths used by a processor (or DMA) to transmit signals that contain address information. The width (number of lines) of the address bus determines the address space.

Address Space - The number of addresses that can be uniquely specified. The maximum amount of memory (usually in bytes) a processor can handle. (e.g., 64K bytes).

Addressing Mode - The way the address of an operand is generated by the CPU. Any instruction that reads or writes data to a location generates the address of the location through an addressing mode.

Architecture - A representation of the hardware and software components of a system and their interrelationships, considered from the viewpoint of the whole system, as opposed to that of a single part.

Arithmetic Logic Unit (ALU) - That portion of the computer central processing unit which performs the Boolean and arithmetic operations on data inputs. The accumulator acts as the storage location for the results of these operations. Typical elements consist of an adder and a shift register.

Arithmetic Processor - A special purpose processing unit which is designed to operate in conjunction with a central processing unit. Its functions are, however, limited to the arithmetic functions.

Arithmetic Shift - A distinction important for shift-right operations where the most significant bit is held constant. This preserves the sign of the original operand. An Arithmetic shift-left is the same as a logical shift.

Assembly Language - The specific mnemonics which are supplied as data to an assembler program to produce the machine code for a specific computer.

Asynchronous Event - An event which is not coordinated with or controlled by the computer clock.

Auto-Increment Addressing - An addressing mode whereby the register used to generate the effective address (usually an index register) is incremented before (pre-) or after (post) an instruction that uses auto-increment addressing is executed.

BCD - Acronym for binary coded decimal. A four bit code that represents the numerals 0 thru 9. Since four bits can denote sixteen distinct symbols. Combinations of 1010 through 1111 are not used. Some processors can perform arithmetic directly in BCD thus eliminating decimal to binary conversions.

Backplane - A set of electrical signal paths which allow interconnection of several functional units of a computer system. The paths are normally construted on a printed circuit board or card, and include connectors for printed circuit boards which contain the functional units. Several defacto standards exist in the industry for these paths. Sometimes referred to as a mother board.

Base Page - Usually the first page of memory usable for data. The page in which base page addressing occurs.

Base Page Addressing - A short addressing mode where all addresses are actually offsets within locations of memory called the base page. This is usually a faster addressing mode since part of the address is implied. Most machines fix the base page to a physical address. See Base page register.

Base Page Register - A base register whose contents is the first address of the base page.

Base Register - Similar to an index register. A register whose contents is an address to which offsets are added to generate the effective address of an operand. Base registers usually contain the first address of an array or data segment.

Bidirectional Line - A line coming from a device on which the device can either insert voltages (output or write) or sense voltages (input or read).

Bipolar - (1) Exhibiting plus and minus voltages with respect to a reference. A bipolar power supply. A bipolar signal. (2) A transistor type that amplifies current and is generally faster than field-effect transistor. Bipolar denotes the devices use of both minority and majority current carriers.

Bit - Binary Digit. The most primative piece of information. Can either be a logical 0 or a 1.

Bit Slice Architecture - A high performance CPU that is composed of several smaller, high speed processing elements called bit slices. A bit slice element can perform basic arithmetic and boolean operations on part of the data word, typically four bits. The slices are connected together in such a way to allow arbitrary CPU word lengths, typically 16, 32, 48 or 64 bits wide (4, 8, 12, and 16 four bit slices, respectively).

Bit Slice Processor - A processor which is composed of bit-slice elements. This type of processor has the ability to change its machine code and word length within the bounds of bit-slice capability.

Bitmap Graphics - A raster scan graphics mode where each dot (pixel) on the screen directly corresponds to the state of a bit in memory. (e.g. 1=on, 0=off).

Board - A "printed circuit" board composed of one or more layers of copper and some insulating material so constructed as to perform some defined function when populated with integrated circuits and discrete devices. Alternative names include card, pc board, printed circuit card and printed wire assembly.

Bootstrap - Usually a small program located in the memory location which the central processing element vectors to when power is supplied. Its purpose is to provide enough logic to allow the CPU to control a load into memory, from a storage device, some larger program, thereby minimizing the amount of memory used for permanent program storage.

Breakpoint - A pointer to a memory location that is used for debugging. When a program under test executes the location to which the breakpoint is pointing, processing is interrupted and control is returned to a monitor program.

Buffer - (1). An array of memory locations used for the temporary storage and accumulation of data. (2). A circuit which is used to provide additional drive (fan-out) or isolation.

Buffered I/O - Input/output operations which store/retrieve data from a buffer in memory. Double buffered I/O allows increased I/O throughput since the processor can acquire data from one buffer while an I/O device is loading into another.

Bus - A set of electrical signal paths which perform a definable function or related functions. Microcomputer buses include address, data, control and power busses.

Bus Driver - A logic circuit that is a buffer specifically designed to increase data or address drive (fanout) capability. These are found in medium-large systems since a microprocessor bus fanout is usually quite low for power dissapation purposes.

Bus Tranceiver - A logic circuit used to increase bus drive and provide isolation. This is similar to a bus driver but is bidirectional. A control line determines which direction the information is flowing.

Byte - A collection of eight bits. Able to represent $2^8 = 256$ combinations. A byte has also been used to denote collections of other than eight bits.

CAM - Acronym for content addressable memory.

CCD - Acronym for a charge coupled device. A device, usually a memory device, which uses the storage of electronic charge on a capacitor as the information storage mechanism.

CMOS - Acronym for complimentary metal oxide semiconductor. A process which produces integrated circuits characterized by low power consumption. A CMOS gate contains both NMOS and PMOS transistors.

CPU - Acronym for central processing unit.

CRC - Acronym for cyclical redundancy check.

Cache - A very high speed memory that buffers between the CPU and main memory. A cache can provide increased throughput by prefetching instructions and data, and by storing intermediate variables.

Card - An alternative name for a printed circuit board.

Carry - A bit or flag that results during an addition when the magnitude of the sum is greater than the precision of the register in which the addition was performed. Left-shifts will often shift the MSB into the carry bit.

Central Processing Unit (CPU) -The portion of the computer which contains the arithmetic logic unit, its support circuitry, and may include a coprocessor or memory. It represents a functional unit of the computer architecture and can be composed of a microprocessor, microcomputer, or bit-slice elements.

Clock - A fixed frequency signal, or set of signals, which provide the synchronization for a computer and its elements.

Clock Cycle - The period of the computer clock, frequently used as a frequency independent measure of instruction execution time.

Compaction - The reallocation of programs in memory to make them contiguous.

Compiler - A program which translates a high level language program (source) either into intermediate form or directly to machine code (object).

Complement - To take logical opposite or to be in the opposite state. The complement of 1 is 0, High is Low, 0110111 is 1001000, etc.

Computer - A machine capable of performing computational functions in a prescribed manner, frequently in a synchronous, serial manner. Typical elements consist of a central processing unit, memory, and I/O.

Concurrent - An operating system term denoting simultaneous execution. Two or more tasks that are executing (active) at the same time are called concurrent.

Concurrent Programming - A programming technique that employs simultaneous execution of two or more program paths. Methods for synchronization and communication between the concurrent paths are required.

Conditional Branching - A program branch that occurs only if a certain condition (e.g. zero, overflow, etc) is true. Otherwise the next program step is executed.

Content Addressable Memory (CAM) - An intelligent memory device that returns the address of the location that contains specified contents. This involves the simultaneous comparison of all locations. Conventional memory returns the contents of a specified address.

Context Switching - The operation involved in saving the state of a currently operating task and the restoration of the state of a previously operating task. This usually requires the saving of all CPU registers and sometimes the return address.

Control Bus - The set of electrical paths which carry control information internal to a computer or central processing unit. Examples include interrupt signal paths, read/write signal paths, and enable signal paths.

Control Character - A non printing character code (usually ASCII) which is treated by an I/O device as an instruction rather than data.

Controller - A computer interface which provides control signals to a peripheral device. It may or may not provide a data path as well.

Coprocessor - A processing unit which acts in parallel with the central processing unit and uses the same instruction stream as the central processing unit, but otherwise independent.

Cycle Stealing - A way of obtaining control of an address and data bus by only using it when the processor is in a non-bus cycle or a wait-state.

Cycle Time - (1) The time it takes the processor to execute its simplest instruction. (2) The time it takes a memory to recover from an access. The minimum time between memory accesses. See access time.

Cyclical Redundancy Check (CRC) - An error detecting and correcting technique that is usually employed as tag bits to disk records.

D/A Conversion - Refers to Digital to Analog conversion. The process or method of producing an approximation of an analog value from a digital input.

D/A Converter - A device which performs digital to analog conversion, frequently an integrated circuit.

DIP - Acronym for dual in-line package. An industry standard IC package that contains two rows of pins, each pin placed on 0.1 inch centers. The spacing between the rows can vary. DIPs typically come in 8, 14, 16, 18, 20, 22, 24, 28, 40, and 64 pin sizes.

DMA - Acronym for direct memory access.

DOS - Acronym for disk operating system. An operating system that resides on, makes accesses to, and manages files on a disk. The term usually refers to a floppy disk based system.

Daisy Chain Priority - A priority arbitration scheme whereby contending devices are strung together in a serial fashion from high or low priority. When the highest priority device is inactive it passes control to the next device, and so on down the chain. This is usually a slow scheme due to the propogation delays in each device along the chain.

Data bus - The computer bus which contains the signal paths for information which the central processing unit assumes to be data. It is this bus for which the various classes of microprocessors, microcompuers, and bit-slice processors are named (i.e. 4-bit, 8-bit, 16-bit, etc).

Debugger - A small program, usually implemented in ROM, that provides examination and alteration of machine registers. It is usually used for debugging programs that are written in assembly or machine language. Provisions for breakpoint insertions are usually found. This is a very low level way to debug a program. Similar to monitor.

Decrement - To reduce a numerical value. A variable is decremented if its value is less than its previous value. A common decrement operation is to subtract one.

Device Driver - Synonym for device handler.

Device Handler - A program or subroutine which interfaces an I/O device to nigher level programs. A handler is specially tailored to a particular device.

Die - A piece of silicon that is the actual integrated circuit. Die are usually placed in packages and connected via wire bonds from the die (on bonding pads) to leads that emanate from the package.

Direct Execution Machine - A sophisticated processor whose instructions are directly mapped from statements in a High Order Language. These machines are usually object oriented. (e.g. the Intel iAPX432 is a direct execution machine for Ada. It is also an object oriented machine).

Direct Memory Access (DMA)- A high speed data transfer method whereby data are directly read (or written) from memory without processor intervention. A processor responds to a DMA request by floating the address and data lines, thereby relinquishing control of the memory to the DMA device.

Disk - Mass storage media which uses one or more disks whose surfaces use a magnetic coating.for information storage.

Distributed - In reference to a system, the segmentation of functions, tasks, or processes into segments which are functionally and may be physically separated.

Distributed Processing - Processing accomplished by a system in which the computing functions are dispersed among several physical computing elements. Examples include the floating point processor of a computer and the navigation and flight controls computers of an avionics sytem.

Dynamic Logic - Logic that requires continuous clocking to preserve valid internal states. Logic levels are internally stored as charges, which are dissapated after short periods of time. Many high density NMOS circuits use dynamic logic, including dynamic RAMs and microprocessors.

Dynamic Memory - A type of read/write memory which is volatile and requires periodic refreshing in order to maintain the information stored.

EAROM - Acronym for electrically alterable read-only memory. A non-volitile memory whose contents can be changed by the application of electrical pulses. See EEPROM.

ECC - Acronym for error checking and correcting. ECC refers to computer hardware which checks for errors and corrects them as part of the memory access.

ECL - Acronym for emitter coupled logic.

EDAC - Acronym for error detecting and correcting. Refers to a data encoding/decoding technique, memory system or mass storage drive that can detect and correct bit errors. See Hamming Code, Cyclical Redundancy Check.

EEPROM - Acronym for electrically erasable PROM. A PROM that can be erased by the application voltage pulses. The PROM can then be reprogrammed.

EPROM - Acronym for erasable programmable read only memory. EPROM Refers to a PROM that can be erased, usually by the application of ultra violet light. It can then be reprogrammed.

Effective Address - The absolute location that is the result of addressing mode calculations. The address that is actually generated by an addressing mode which refers to a specific location in memory.

Emitter Coupled Logic (ECL) - A high speed logic family used in large scale computers and some bit-slice processors. ECL Provides high speed capability and operates between 0 volts and -5 volts.

Emulator - (1) A program which emulates or models a device, particularly a processor. (2) A collection of hardware which substitutes for a device by maintaining the proper input/output relationships. See In-Circuit Emulation.

Entry Point - An address which refers to the first valid instruction in a subroutine or program. Subroutines may have several entry points in which case they are context dependent.

Exclusive - Or (XOR, EOR) - A logic operation that performs exclusive disjunction. (A XOR B) = 1 only if A = 1 or B = 1, but not both, i.e., A and B must be in different states.

Executive - Provides control of the execution of programs in a computer system. It creates and removes tasks, controls the sequence of task execution, handles exceptional conditions arising during the execution of a task (i.e., interrupts and errors), allocates hardware resources among tasks, provides access to software resources, provides protection, access control, and security for information, and provides a means of communicating information among tasks, and between the system and its peripherals.

FIFO - Acronym for first in first out. A memory buffer technique where data (words) are shifted through during writing. The first word written is the first word read. Also called an elastic buffer.

FLOPS - Acronym for floating point operations per second. The number of floating point operations (usually single precision multiplies) that can be performed in one second.

Fanout - A specification of a circuit's output drive capability. A fanout of N means that the circuit can drive N inputs of a specified logic family (usually TTL).

Field Programmable - A ROM or PLA that can be programmed (in the field) by the application of a sequence of voltages which selectively open internal links (usually nicrome). See Mask Programmable.

Firmware - Software residing in a read-only memory device.

Fixed Point - Synonomous with Integer.

Float - Refers to the high impedence state of a tri-state line. Once a line is floated, either a 0 or a 1 can be asserted by another device on the line.

Floating Point Processor - A specialized coprocessor or peripheral which performs floating point operations. Transendental calculations may also be supported.

Floppy disk - A disk mass storage mechanism which uses flexible, removable discs as the storage media. It is in common usage as mass storage in microcomputer systems.

Full Adder - A circuit that adds two binary numbers plus a carry input signal.

Full Duplex - A communication system in which terminals are capable of simultaneously transmitting and receiving (i.e. simultaneous transmissions in two directions are possible).

Functional Throughput Rate (FTR) - FTR is a figure of merit for throughput of a given chip, FTR = gates/chip x clock rate (units of Gate-Hertz).

GPIB - Acronym for general purpose interface bus. GPIB is a communication bus standard characterized by a parallel data format. This standard is also known as the IEEE 488 standard. Its most common use is for interconnecting computer based test equipment.

Gate Array - A collection of several hundred or thousand logic gates (usually implementing NAND or NOR logic) that are regularly placed on an IC but whose interconnect is specified by the user.

Gibson Mix - An instruction time weighting mix which allows performance comparison of different processors. The times taken by the processor to execute various types of instructions are multiplied by weighting factors and then summed. The result is in thousands of operations (instructions) per second (KOPS).

Ground Loop - A difference in ground potentials at two different points in a circuit or system. These are caused by large current flows along ground paths thus creating a voltage drop.

HDLC - Serial data communcation standard.

HLL - Acronym for High Level Language. Synonomous with HOL.

HMOS - Acronym for a new high-performance MOS process used in the manufacturing of VLSI components, particularly microprocessors and memory components.

HOL - Acronym for high order language.

Half Adder - A circuit that performs the addition of two binary numbers, but cannot accept a carry input.

Half Duplex - Operation of a data transfer system in one direction at a time over a single bidirectional channel.

Hamming Code - A special type of error detecting and correcting code that is usually appended to transmission data.

Handshake - A set of signals or a protocol between a receiver and transmitter. Transmission does not occur until the receiver has acknowledged the transmitter's request to transmit. This is usually used to provide syncronization between systems and to resolve multi-user contention for a common resource.

Hardware - Physically existing components, as contrasted to software.

Hardwired - A function or condition which is set by the interconnection of the hardware and cannot be altered by software.

Hexadecimal - An abbreviation system that represents binary numbers in groups of four bits. (base 16). It includes 16 symbols, 0 thru 9, A thru F. E.G., A_{16} = 1010_2, 5_{16} = 0101_2, and FE_{16} = 11111110_2.

High Order Language (HOL) - A programming language which is machine independent and allows expressions and program structures which translate into many machine instructions. The translation to machine code is performed by a compiler. HOLS are in contrast to assembly languages which are machine-dependent and require a one-to-one correspondence between assembly language statements and machine code. Examples of HOLS are FORTRAN, PASCAL, COBOL, JOVIAL, ALGOL, and Ada.

Hybrid - (1) A circuit built on a ceramic substrate that contains a die or dice and discrete, unpackaged components such as resistors, capacitors, and transistors. See monolithic. (2) A circuit that mixes analog and digital components.

I^2L - Acronym for integrated injection logic.

IC - Acronym for integrated circuit

I/O - Acronym for input/output.

ICE - Acronym for in-circuit emulation.

ISA - Acronym for instruction set architecture.

Increment - To increase a numerical magnitude. A variable is incremented if its value is more than its previous value. A common increment operation is to add one.

Index Register - A register whose contents can be added automatically to an address field contained in an instruction to generate a particular address.

Input/Output - Hardware and software required to communicate with an external device.

Instruction - A pattern recognized by a processing element which causes the element to perform a defined operation.

Instruction Cycle - The set of logic states contained in execution of an instruction. Alternately, the number of clock cycles needed to execute an instruction.

Instruction Set - The set of all machine codes that a particular processor can recognize. The instruction set usually is presented in assembly language mnemonics rather than machine code binary.

Instruction Set Architecture (ISA) - The attributes of a digital computer as seen by a machine (assembly) language programmer. ISA includes the processor and input/output instruction sets, their formats, operation codes and addressing modes: memory management and partitioning if accessable to the machine language programmer; the speed of accessable clocks; interrupt structure; and the manner of use and format of registers and memory locations that may be directly manipulated or tested by a machine language program. This definition excludes the time or speed of any operation, internal computer partitioning, electrical and physical organization, circuits or components of the computer, manufacturing technology, memory organization, memory cycle time, and memory bus widths.

Integrated Injection Logic (I^2L) - A bipolar logic family typified by low power consumption and high-density. Speed is programmable through the selection of current sources.

Integrated Circuit (IC) - The integration, i.e., collection of, more than one device on a single piece of silicon.

Integration - The process of combining two or more functions together. For example integrated circuits combine circuit elements together on a single substrate.

Interface - (1) A device which translates external signals into formats useable by a processor. (2) The mechanism which provides the connection between two functional elements. It includes both hardware and software mechanisms.

Interpreter - A program that executes a sequence of commands that are not in machine code, i.e. it must interpret them. Interpreters can execute commands typed in from a keyboard or programs, e.g. BASIC. Interpreters for programs are slow because of the translation required. See Compiler.

Interrupt - A signal to a processing element which causes a change in the direction of process flow.

Interrupt Controller - A device which attaches to a processor bus that arbitrates among input lines representing interruptions from other devices. The controller then interrupts the processor and provides it with information necessary to identify the interrupting device.

Interrupt I/O - An I/O device servicing technique where the device interrupts the processor when it requires servicing rather than the processor polling the device when it needs servicing.

Interrupt Masking - The ignoring of certain interrupt requests by the processor. The processor sets a bit or a "mask" to deselect the interrupts to which it does not want to respond. This may be necessary during time critical program segments that must not be interrupted. See Non Maskable Interrupt.

Interrupt Service Routine - A subroutine that is executed in response to an interrupt. There is usually a special service routine for each interrupt supported by the system. Since interrupts are usually issued by an I/O device, service routines typically are device handlers.

Interrupt Vector - A direct or indirect pointer that identifies the interrupting device. The vector is provided by the device, by an interrupt controller, or by the processor itself in the case of high priority or internal interrupts. A direct vector gives the address of the service routine. An indirect vector just identifies the device.

Inverter - The basic logic circuit which converts high voltages on its input (corresponding to a 1) to a low voltage on its output (corresponding to a 0), and vice-versa. The inverter implements the logical complement of a signal.

In-Circuit Emmulation (ICE) - The replacing of the microprocessor unit with a device which emmulates its signals. ICE is typically used for debugging because it allows greater access to the internal register.

Jump - A transfer of program control to an absolute location. The program counter contents is replaced with the new address. Program execution continues from this new address. Equivalent to a FORTRAN GOTO statement.

K - (1) Abbreviation for x1024, e.g., 64K = 65,536. 1024K = 1024^2 (1 MEG)
(2) Abbreviation for x1000, e.g, 1 KOP = 1000 Operations per second, 1 KOhm = 1000 ohms.

LIFO - Acronym for last in first out. A memory buffer technique where data (words) are continually "pushed down" during writing, and "popped off" during reading. The last word written is the first read. Also called a stack.

LSB - Acronym for least significant bit or least significant byte. The low order (least significant) bit or byte in a word. A change in the LSB results in a smaller change in the magnitude of a word than a change in the MSB.

LSI - Acronym for Large Scale Integration.

Large Scale Integration (LSI) - A reference to the level of complexity of an integrated circuit, usually between 1000 to 10,000 devices on a single integrated circuit.

Latency - Refers to delay (latency) that is experienced when accessing a disk due to the time it takes the disk to go through a revolution. Can also refer to the time it takes a processor to respond to an interrupt. In general, any delay due to information flow and reaction time.

Linked List - A collection (or list) of data where each data element or groups of elements has a pointer (link) to the next element in the list. It may also contain a pointer to the previous element (double linked list). The processes of going through a list is called traversing. A linked list makes the addition and deletion of elements easier.

Linker - A program which joins together separately assembled or compiled subprograms. The linker must resolve all references that are made among the separately compiled subprograms since this was not possible during the compile.

Loader - A program which copies a program from a mass storage device to memory locations in memory. This process may involve final resolution of [illegible] references from one subprogram to another

Lock - An action taken by a processor in a multi-processor system to gain control of an area of memory or a bus. The lock typically acts to [illegible]. See Semaphore.

Logical Address - The address of a location in logical memory that is associated with the program and is subsequently mapped to a physical address (actual machine location). The process of calculating the physical address is called address binding, mapping, or translation.

Logical Shift - A shift where zeros are shifted into the most significant bit (right-shift) or least significant bit [illegible].

Lookahead - (1) An arithmetic circuit which [illegible] interstage carries. (2) A high speed [illegible] beyond the current instruction for [illegible].

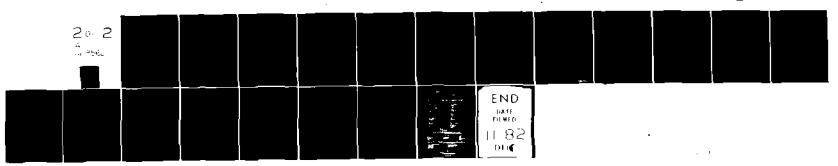
MNOS - Acronym for metal nitride [illegible] memory devices to make them [illegible] sustained, however.

MOS - Acronym for metal [illegible] an integrated [illegible].

MSB - Most significant [illegible] bit or byte [illegible] byte [illegible].

AD-A119 562 ADVISORY GROUP FOR AEROSPACE RESEARCH AND DEVELOPMENT--ETC F/G 9/2
DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTRO--ETC(U)
JUL 82 L J URBAN

UNCLASSIFIED AGARD-AR-178 NL



MSI - Acronym for medium scale integration.

Machine Code - A collection of machine instructions. The actual binary pattern that represents the program (code) the CPU executes. See Machine Instruction.

Machine Cycle - The collection of clock cycles which perform a set function, such as a memory cycle, or an I/O cycle.

Machine Instruction - The binary pattern that represents the operations the CPU can perform. Assembly instructions are the Mnemonics for machine instructions. See Microinstruction.

Macro - A command, instruction, or literal which is used as an abbreviation for a sequence of commands, instructions, or literals. Usually used in programming languages as an abbreviation for instruction sequences commonly used in a program.

Macro-Assembler - An assembler that supports macros, i.e., it allows abbreviations for instruction sequences to be defined by the user.

Magnetic Bubble Memory - A memory technnology which uses magnetic domains for information storage. They are used as mass storage devices analogous to disk systems.

Mask Programmable - Refers to a ROM whose bit pattern (programming) is established when the chip is manufactured. See Field Programmable.

Maskable Interrupt - An interrupt that can be disabled and ignored under processor control.

Mass Storage - A device such as a disk or tape that can store large quantities of data. Nonvolatility and relatively slow access times are usually implied.

Master - The element in a computer or other bus system which controls information transfer on the bus.

Medium Scale Integration (MSI) - A reference to the level of complexity of an integrated circuit, usually between 100 and 1000 devices on a single integrated circuit.

Memory - The functional element of a computer which stores instruction and data information for use by the processing elements.

Memory Interleaving - A technique where several memory words are simultaneously accessed as a contiguous block to increase memory access time beyond that of a single memory. This only works for sequential accesses (instructions). Memory accesses are slightly staggered then multiplexed onto a bus.

Microcode - The binary information which provides gate level control in a central processing unit. Each CPU instruction causes execution of a segment of microcode, wnich in turn is composed of a collection of microinstructions.

Microcomputer - A single chip containing all microprocessor elements plus some internal RAM, ROM, and I/O. Also can be a board configured with a microprocessor, RAM, ROM, and I/O.

Microinstruction - An instruction which is executed internal to the CPU as a result of an externally fetched machine instruction. A microinstruction contains many bits for fine, but primative control of internal CPU registers and busses. Several microinstructions are executed during a typical single machine instruction execution.

Microprocessor - A single chip comprised of an arithmetic logic unit, address and data bus, control logic, and an associated instruction set architecture.

Microprogram - The program executed by the central processing unit when it receives an instruction. The microprogram is composed of microcode.

Microprogram Sequencer - The element of the central processing unit which causes sequential execution of the CPU microcode. Usually a single device in bit-slice processors, and integral to monolithic microprocessors.

Minicomputer - A computer of larger capacity or capability than the microcomputer. Advances in integrated circuit technology have blurred the distinction between the two.

Mnemonic - Symbolic representation of an instruction, e.g., ADD, SUB, LDA, etc., used in assembly language programming.

Modem - A term for a modulator/demodulator. A device used to convert digital information to analog form (modulator) and the analog form to digital form (demodulator).

Monitor - A small program that enables primative programming (usually in machine code) and examination and alteration of CPU registers. Similar to a debugger.

Monolithic - Denotes a circuit that is implemented on a single piece of silicon, called a wafer. See Hybrid.

Multi-Tasking - Refers to an operating system or executive that can support simultaneous execution of tasks.

Multi-User - Refers to an operating system or executive that can support several users simultaneously. Each user has a set of tasks which are associated with it. A multi-user system is therefore a hierarchical multi-tasking system.

Multibus - A commonly used microprocessor bus specification developed by the Intel Corporation and used as a defacto industry standard.

Multiplex - Any method which provides a means of integrating a multiple number of signals into a single signal line or path. Also the act of performing the integration.

Multiplexer - A device which multiplexes signals.

Multiprocessing - A term referring to the use of more than one processing element on a given bus. This may be more than one CPU internal to a computer to multiple computers connected by a communications bus.

NAND - The complement of the logical AND operation.

NMI - Acronym for non maskable interrupt.

NMOS - Acronym for N-channel metal oxide semiconductor. A MOS process which produces N-channel devices. NMOS is a very dense semiconductor. As such, it is the technology usually employed in microprocessors.

NOR - The complement of the logical OR operation.

Negate - The arithmetic equivalent of a logical complement. In most microprocessors this involves the complement of all bits in a word to which a 1 is then added. See Two's Complement.

Nibble - A collection of four bits. One half of a byte.

Non Maskable Interrupt (NMI) - A special, high priority interrupt that cannot be disabled (masked) by a processor. NMI is usually used to warn the processor of a fault condition or impending disaster (e.g. loss of power). See interrupt masking.

Non-Volatile - Refers to a logic circuit, especially a memory, that will retain internal states even when no power is applied to the device.

Normalization - A process performed after a floating point operation that shifts the mantissa left while decreasing the exponent until a 1 is found.

OR - A logic operation that performs disjunction. (A or B) = 1 only if A = 1 or B = 1, or both. Symbolically represented as A+B.

Object - (1) Synonym for Object Code. (2) An arbitrary collection of bits. See object oriented machine.

Object Code - The machine instructions (binary patterns) that are produced by a compiler or assembler and can be directly executed by the processor.

Object Oriented Machine - A machine which deals with objects (arbitrary collection of bits) rather than words. Addresses are generally not seen by the progammer, only references to objects. This is an advanced architecture and is suited particularly for High Level Language programming.

Octal - An abbreviation system that represents binary numbers in groups of three (base 8). It includes eight symbols 0 thru 7. e.g. $6_8 = 110_2$, $17_8 = 001111_2$.

On-Chip Clock - A term used to identify microprocessors that contain all the necessary circuitry to generate the clock signal except for an external crystal or RC network.

Opcode - That part of a machine instruction that represents the specific operation to be performed. (The rest of the instruction may an operand (s)).

Operand - An entity such as a byte, word, or address that is operated upon during the execution of an instruction (the opcode).

Operating System - The program of a computer which provides the functions of a monitor and also provides software tools such as compilers, editors, and file maintenance. An operating system provides the input/output interface from a program to the actual computer environment.

Opto Isolator - An electrical isolation circuit for the transfer of logic signals. It employs a light emmitting transmitter and a photo sensitive receiver. Several thousand volts of isolation can be obtained.

OR-Tied - (Also Wired-OR) A logic circuit operating feature whereby the logical ORing of digital signals is realized by physically connecting the signals at a common node. In this configuration any independent signal can drive the node to a particular logic level, hence the OR function is accomplished. This is often realized by connecting outputs such that a line will be low if any output on the line is holding it low. Otherwise the line is held high through a resistor, called a pull-up, attached to the positive supply.

Overflow - A condition that occurs when an operation results in quantity beyond the representation of the register in which it was performed. This condition is a sign-sensitive form of a carry.

Overlay - A section of code or data that is brought in from mass storage and layed or written over a previous set of code. This is used when a program requires more memory than is physically available.

P-Code - A pseudo instruction or collection of pseudo instructions.

PIA - Acronym for peripheral interface adapter. A general purpose interface or I/O IC that allows the processor to read and assert parallel lines to peripherals and the outside word.

PIC - (1). Acronym for position independent code, or (2). peripheral interrupt controller.

PIO - (1) Acronym for parallel input/output, or (2) programmed input/output.

PLA - Acronym for programmable logic array. A regular structure of AND and OR gates that allow mapping of "sum of product terms" (i.e. combinatorial expressions) of inputs into the outputs. This is a regular structure intended to replace combinatorial circuits normally implemented in random logic.

PMOS - Acronym for P-channel metal oxide semiconductor. For equivalent transistors, PMOS operates at a lower speed (typically 1/2) than NMOS in silocon. Thus NMOS processes are preferred.

PROM - Acronym for programmable read only memory.

Page - A contiguous block of memory of standard size which is aligned to an address boundary that is a multiple of this standard size. A typical page size for microprocessors is 256 bytes.

Page Register - A register whose contents is the first address of a page of memory. References are made with respect to this register (and thus within the page to which it is pointing) using paged addressing.

Paged Addressing - An addresing mode that decreases execution time and code size by using a page register to supply part of the address. See Base Page Addressing.

Parallel I/O (PIO) - The input and output of a computer which transmits or receives more than one bit at a time. Usually these types of transfers are done on 8-bit bytes or 16-bit words.

Parallel Processor - A processing element which performs its operations at the same time as the central processing unit. The instruction flow may be either the same as or independent of the CPU instruction stream.

Parity - Any encoding technique which adds extra information to the digital data being processed or transferred to facilitate error detection. The typical application is single bit parity which may be even parity or odd parity. The parity bit is set to make the number of logic 1 bits in the monitored data even for even parity or odd for odd parity. In either case, even multiples of bit errors will not be detected.

Peripheral - (1) On the system level, the printers, disks, and I/O devices which surround the computer. (2) On the computer level, the circuits which control the system level peripherals.

Physical Address - An actual location or address to which a memory cell or device responds. The true location of a memory cell with respect to the hardware. See logical address.

Pipeline Register - A register which is used to buffer information between two pipeline stages. Typically found in instruction paths holding the present instruction while the next instruction is generated.

Pipelining - The separation of processing into several stages. Each stage is always active and passes results to the next stage at regular intervals. This allows throughput for repetitive calculations to be much higher than for a similar network without pipelining.

Pointer - A variable whose contents is an address. i.e., it 'points' to a location in memory. Also, a variable whose contents is an offset that forms an address when added to an index or base register (e.g. a pointer to an array).

Polled I/O - A peripheral device service technique where the processor determines the state of the peripheral by continuous interrogation. This is in contrast to an interrupt technique whereby the peripheral interrupts the processor when it has finished an operation or detected an error.

Port - A functional element of a computer which acts as a I/O channel for peripheral device(s).

Position Independent Code (PIC) - A program that contains no absolute references to memory, only relative ones. Thus, the program can be located at any physical address and still operate properly.

Power On Reset - A dedicated non maskable interrupt which restarts program execution at a predefined location at the detection of a power up or restart signal.

Priority - The relative importance assigned to a computer task or device for the purpose of determining the order in which process elements will be performed.

Processor - A system which performs a sequence of operations or a device which provides a mechanism for performing a set of arithmetic or logical operations which evaluate an input and yields some result.

Program Counter - A register whose contents is an address that points at the next instruction to be executed.

Programmable Read Only Memory (PROM) - A variant of ROM which is field programmable.

Protocol - A set of terms or definitions which determine the structure of information to be transfered between two or more points in a system.

Pseudo Instruction - An instruction that corresponds to a fictitious machine (P-machine). An interpreter translates the Pseudo-Instructions into machine instructions. This approach is used to make software machine independent, only requiring the relatively simple interpreter.

Pseudo Static RAM - A form of dynamic RAM that contains transparent on-chip refresh.

QUIP - Acronym for quad in-line package. A high density packaging technique used to replace DIPs of 40 or more pins. In this case four staggered rows of pins are employed.

Qualify - To take the conjunction (AND) of a signal with another signal called the qualifier. B qualifies A is the same as B AND A.

RAM - Acronym for random access memory.

ROM - Acronym for read only memory.

RS-232 - An Electronic Industries Association (EIA) standard for serial communication protocol used in terminal and computer communications.

Random Access Memory (RAM) - A type of memory which allows alteration of the information stored within it. A memory which can be written to or read from. Often referred to as read/write memory.

Random Logic - Refers to the lack of a regular organization or structure in a logic network. It does not imply a lack of determinism.

Read - With respect to the processor, the action of acquiring the contents of a particular address (memory location) to be used within the processor.

Read Only Memory (ROM) - A non-volatile memory which allows only reading of the information stored within it. Some versions have a mechanism by which they may have their contents altered but they are not capable of being written by the CPU.

Read-Modify-Write - A single instruction that simultaneously reads a memory location and replaces it with a new value. See semaphore.

Real Time Operating System - An operating system that is tailored to have small latency with respect to interrupt requests. See executive.

Reentrant - An attribute of program that allows several tasks to be simultaneously executing the same code. This means that the code cannot be self-modifying and all data references occur with respect to a stack or a variable data pointer (task supplied).

Refresh - With respect to dynamic logic, especially dynamic RAMs, the process in which the charges which represent the logic states are renewed. In a dynamic RAM this requires periodic read or refresh cycles for each internal memory row or column.

Relative Addressing - An addressing mode where references to memory or operands are made relative to some register, (usually the program counter for branch instructions). The register provides an absolute address pointing to memory. An instruction employing relative addressing adds or subtracts displacements to this pointer.

Restart Vector - A special case of a direct interrupt vector that contains the address (entry point) of a routine to be executed after the CPU receives a reset signal. Restarts are non-maskable and are the highest priority interrupt.

Return Address - The address of the next instruction to be executed in the program that called a subroutine, once the subroutine has completed. Return addresses are stored on a stack in most microprocessors.

S-100 - A specification for the buses of a computer, which has become a defacto industry standard.

SOS - Acronym for silicon or sapphire. A technology and process which uses a sapphire substrate rather than a silicon substrate for building up the integrated circuit.

SSI - Acronym for small scale integration.

Scheduler - That part of an operating system or executive that decides the order in which tasks will be executed.

Segment - A logical or physical grouping of code or data memory. Some machines support segmentation which is a general form of paging. That is, the segments (pages) can be of variable length and location.

Semaphore - A bit or word that indicates whether a system resource (e.g. peripheral) is being used by a task in a multi-tasking or multiprocessor system. A read-modify-write or lock is one of the only ways to manage semaphores without the risk of two tasks claiming control of a device.

Serial I/O - An input/output function which transmits or receives one unit per clock cycle. A unit may be a single information bit or multiple bits.

Shift - An operation on a word or byte which replaces each bit with the contents of the bit from the immediate right (shift left) or from the immediate left (shift right).

Shift Register A register capable of performing shift operations.

Single Board Computer - A computer usually a microcomputer, which contains its functional elements on a single printed circuit board, i.e., CPU, memory, and I/O.

Single Chip Computer - A computer, usually a microcomputer, which contains its functional elements on a single monolithic substrate.

Single Step - A process used in program debugging where instructions are executed one at a time, allowing inspection of registers and busses between each instruction.

Slave - A subordinate device controlled by a master and communicates over some medium.

Small Scale Integration (SSI) - Acronym for small scale integration. SSI usually refers to an integrated circuit which has fewer than 100 circuit elements on a single monolithic substrate.

Software - The collection of programs and routines associated with a computer as contrasted to hardware.

Source Program - A higher level representation of a program that is not directly executable by a machine. It must first be translated into machine instructions to create the object program, either by a compiler or an assembler.

Stack - A buffer or segment of computer memory which is reserved for use by the central processing unit. Normally used for saving state information resulting from context switching.

Stack Pointer - The register in the central processing unit which contains an address within the stack.

Standard Bus - A specification for bus signals which has become a defacto industry standard.

Static - A device that does not require refreshing of internal logic states. I. E., it contains no dynamic logic.

Status Register - A register internal to the central processing unit which maintains information on the results of the central processor operations.

String - A linear collection of bytes whose contents are interpreted as characters, usually ASCII.

Strobe - A signal line that enables a gate or register. Typically refers to the signal that causes a register to store the data present at its input.

Synchronous Event - An event which occurs in conjunction with the computer clock and can be timed relative to the clock.

TASK - A program or subroutine. Any autonomously executing set of code. See multi-tasking.

TTL - Acronym for transistor transistor logic. A bipolar logic family which has certain defined electrical characterists and mechanisms for implementing logic functions. These electrical characteristics are the defacto definitions for pins on most devices, regardless of the logic family used to implement the device.

Terminal - A functional element which provides the interface for entering information to a computer or receiving information from a computer.

Transparent Refresh - A technique where refresh of dynamic RAMs is done during CPU bus cycles in which no memory references are made. Thus the CPU never has to wait for a refresh cycle to complete.

Trap - A system level interrupt structure normally used for processing errors during program execution.

Tri-State - Refers to the output of logic circuit that can have three states; on (1), off (0) or a special high impedence state (z). The z state allows other outputs connected together to establish the state of the line.

Twin Tub - A CMOS process developed by Bell Laboratories that diffusses both n-type and p-type tubs into a wafer of insulating, undoped silicon. This technique promises higher density, higher quality devices.

Two's Complement - A commonly used representation of negative numbers. A number can be changed in sign in two's complement notation by complementing all of it's bits and then adding one, ignoring the carry. e.g. the two's complement of 0000001 (1) is 11111111 (-1).

UART - Acronym for universal asynchronous receiver transmitter. UART is a device which performs serial to parallel and parallel to serial conversions for serial computer I/O.

ULSI - Acronym for ultra-large-scale integration.

USART - Acronym for universal synchronous asynchronous receiver transmitter. A serial communication protocal device which implements synchonous proticol or asynchronous protocol.

USRT - Acronym for universal synchronous receiver transmitter. A device which provides serial-parallel conversion for synchronous serial communications.

UVPROM - A PROM that can be erased by the application of ultra violet light through a transparent window covering the IC. The PROM can then be reprogrammed.

Ultra Large Scale Integration (ULSI) - ULSI usually refers to integrated circuits with a complexity of 100,000 devices per monolithic substrate.

VHPIC - Acronym for very-high-performance integrated circuits. A program of work funded by the UK MOD aimed at advancing the state of the art in Semiconductor technology and applications for Defense. The emphasis is towards developing submicron technology in CMOS, CMOS/SOS, ECL/I^2L and Bipolar CDI, and to identify and develop applications devices.

VHSIC - Acronym for very-high-speed integrated circuits. A United States DOD program designed to stimulate private research and development in microelectric technologies that are currently not emphasized by commercial VLSI programs. This includes very-high-speed, small geometry, radiation hard and complex custom circuitry.

VLSI - Acronym for very-large-scale integration.

Vectored Interrupt - An interrupt to the central processing unit which causes the flow of execution to go to the address contained in the location defined by the interrupt.

Very Large Scale Integration (VLSI) - VLSI usually refers to integrated circuits with a level of complixity of 10,000 devices per monolithic substrate.

Virtual Memory - A mechanism which allows the apparent (virtual) memory size to exceed the actual (physical) memory size. This allows programs to appear to use more memory than is actually resident in the computer.

Volatile - Refers to a logic circuit, especially a memory, that will not retain internal states once power is no longer applied to the device.

Wait State - Extra clock cycles added to a processor to cause it to wait for a slow memory or peripheral, or for the completion of a DMA activity.

Whetstone - A benchmark program developed by the British Government's computer agency. The program is written in FORTRAN and tests the compiler efficiency and processor performance for scientific instruction mixes.

Word length - The length, in bits, of a piece of information, used, frequently as the width of the data bus.

Writable Control Store - A very high speed RAM that can contain user programmed microinstructions.

Write - With respect to the processor, the action replacing the contents of a particular address (memory location) with data generated by the processor.

CHAPTER 4

OPTIONS AND OPPORTUNITIES FOR STANDARDS

by

Professor John T SHEPHERD
School of Electronic System Design
Cranfield Institute of Technology
Bedford MK43 OAL
UK

4.1 GENERAL

Previous sections in this report have examined the state of the art of hardware and software microprocessor based systems. In addition they have looked at the types of system architecture which are available and are currently being implemented. In this section the options and opportunities available for standards are discussed.

It should be emphasised that this document is not intended to suggest definitive standards or even to state categorically that any given standard should be developed. Rather its intention is to focus attention upon the need for standards and to point out areas where opportunities exist for standardization.

As will be seen from the previous sections in this report there is a vast proliferation in hardware and software. When systems are developed they often produce unique hardware and software, such as operating systems, executives, high level languages, etc. Since the life cycle of aircraft systems is at least twenty years from conception, it could be as much as thirty years after the initial design before the systems are finally phased out. This makes it almost impossible to maintain avionic systems in the later parts of their life cycle.

It is often argued that the application of standards can impede development and force the use of obsolete equipment. While these dangers exist, it is possible to produce a standard which is technology independent and which, therefore, does not prevent the introduction of new technology during the life of that aircraft.

To quote some examples of a technology transparent standard, the VOR navigation system initially conceived in the late 30's, was adopted in 1946 as a standard navigational aid by the US, becoming an international standard in 1949. This standard remains the same forty years after the original concept, although a VOR receiver today is rather different from one in the late 1930's or even the late 1950's. Equally the Decca navigation system was introduced in 1944, has chains operating on that standard today although the technology used to implement that system has changed dramatically.

From these examples it will be seen that standards can be produced which are technology independent, and that hardware to meet these standards can be produced by a number of companies. It is precisely this approach that is advocated for standards in microprocessor based systems.

4.2 THE NEED FOR STANDARDS

In considering the need for standards in avionic systems it is worth bearing in mind that every avionic system yet produced has employed some form of standard. At any stage where two units need to interface it is necessary to establish a standard of one form or another so that the designers of the individual units know what that interface needs to be.

With a complex avionic system it is therefore general practice to establish a series of general standards for that avionic system and, using these standards as a framework, produce the detailed form, fit and function specifications for the various elements in the system. In many cases the standards established are unique to that particular system and as such may be produced hastily or without due consideration for the total implications of such a standard.

Such ad hoc standards are typically established to ensure that the various elements in the system are compatible one with another. It follows that if standards are to be established anyway, it is preferable that these standards are properly considered and evaluated by a wider body of people. Once properly established, they can be applied across a number of projects.

In order to achieve this it is essential that such standards are technology independent so that their application does not inhibit the use of advanced technology components where they would benefit the project.

The primary reason for establishing standards is one of cost effectiveness. In the past they have been principally applied to hardware. In general, no matter how large a project is, the economies of scale of procuring large numbers of standard components are such that hardware standardization on a project by project basis is undoubtedly beneficial. This is effectively standardizing on a particular technology base for a project whilst utilising more general technology independent standards to provide the overall requirement. Such an approach applied to both hardware and software undoubtedly increases the cost effectiveness throughout the total life cycle of a project. Today in particular the cost of developing support software is extremely high and, therefore, standards for support software are essential.

Considering each phase of a project in turn the cost benefits accruing from standards are as follows:-

1. DESIGN PHASE

   (a) Since standards are already available, effort does not need to be expended in producing the project standards. Thus system specifications may be produced more quickly and with a high degree of confidence.

   (b) The existence of standards implies that the designer may already have some familiarity with those standards and can incorporate them in his system design.

   (c) The existence of standards mean that the designer can communicate more easily and again reduce the design effort required.

   (d) The existence of standards permits the designer to concentrate on solving the real design problems, rather than being forced to waste time dealing with more routine requirements.

2. DEVELOPMENT PHASE

   (a) The existence of standards means that suitable hardware components should already be available. These components may not be of the latest technology standard but will enable early prototyping to take place.

   (b) The existence of standards means that standard test equipment for hardware and support software tools may be available and, therefore, less development effort will be required to build up the support environment.

   (c) The existence of standards means that a high degree of competence already exists in those parts of the design based upon standards, thus less testing will be required and only the new elements in the system need to be proved.

   (d) The existence of standards means that the development teams are familiar with the standard products and thus can concentrate upon the new areas of design.

3. PRODUCTION PHASE

   (a) The existence of standards means that the economics of scale will apply and that component costs should be lower.

   (b) The existence of standards means that a variety of sources should be available to supply standard equipment.

4. MAINTENANCE PHASE

   (a) The existence of standards means that the maintenance personnel will require less training since they are already familiar with the standards. There are also substantial savings in technical data and technical maintenance documentation.

   (b) The existence of standards reduces the number of test equipments, support software and host computers at a maintenance depot.

Since maintenance costs constitute 70% of the total life cycle of electronic equipment, the aforementioned logistic reasons are more than enough to justify technology independent standards. In practice, the cost benefits also extend to the acquisition phase.

The previous argument also implies that it is worthwhile to establish standards on a national basis and to apply them to a variety of projects. However, if each NATO

country establishes it own standards in isolation, they will probably be incompatible with standards established by other NATO members. This means that the NATO community not only pays more than necessary in establishing standards but that interoperability between NATO forces is also potentially reduced.

It follows that what is needed are international NATO standards, rather than purely national standards. In order to achieve this it is necessary to start considering standards at an early stage. It is of no use waiting until each country has developed its own standards and then attempt to reconcile all of the differing features to produce one international standard. It is far better for NATO countries to collaborate early in producing standards satisfactory to all.

4.3 AREAS SUITABLE FOR STANDARDS

4.3.1 General

In any consideration of standards for microprocessors applied to guidance and control systems, it is necessary to consider both hardware and software. Hardware systems suitable for standards are:-

(a) Interfaces
(b) Processors
(c) Memory systems
(d) Power supplies

while in software it is ncessary to consider:-

(a) High level languages
(b) Executives
(c) Operating systems
(d) System design languages
(e) System description languages.

The following subsections examine each of these areas in detail.

4.3.2 Elements of hardware suitable for standardization

4.3.2.1 Interfaces

The following interfaces are suitable candidates for standardization:-

1. Serial asynchronouse interface
2. Serial synchronous interface
3. Parallel asynchronous interface
4. Parallel synchronous interface
5. Variable data interface
6. Discrete interface
7. Video data interface
8. Fibre optic bus interface.

1. Serial asynchronous interface

A serial asynchronous interface which operates upon a data bus is already widely used in avionic systems. It provides the capability of loosely coupling a federated system of processors and sensors and, because of the loosely coupled nature of such a interface, is ideally suitable for the mission system elements in a total avionic system. Such a system enables the various elements to be redesigned almost independently of the total system concept since the loosely coupled nature of the system means that extremely detailed specifications to cope with synchronisation, etc., are not necessary and the system can be treated in more or less the same way as the earlier generation analog systems as far as conceptual design is concerned.

This technique is already widely used and standards have been produced to cope with it. In particular MIL STD 1553B exists and has gained wide acceptance on both sides of the Atlantic. The UK has recently adopted this standard as DEF Stan 00/18(Pt 2) and NATO is about to adopt it as STANAG 3838.

2. Serial synchronous interfaces

While mission systems are suitable for implementation using a 1553 data bus structure, flight critical systems employing multiplexed redundancy techniques to achieve high integrity require data to be synchronised. Such systems are of necessity more tightly coupled than the mission systems since the system has to be considered as a total entity and designed to ensure that the integrity requirements are achieved. In particular it is essential that operations between the various elements are synchronised, and the data inputs are synchronised between elements and the various sensors providing data. MIL STD 1553B has synchronisation features that could be used for these systems.

3. Parallel asynchronous interfaces

Parallel data transfer is normally used when a high bandwidth data transfer rate is required. Its main application to date is in the internal transfers around a computer system. If tightly coupled multi-processing systems are ever employed in aircraft, then it may be necessary to utilise a parallel transfer mechanism to produce the necessary data transfer rate. Such a system, being tightly coupled and probably incorporating shared memory, would require a high degree of synchronisation. It is, therefore, probably not necessary to consider the development of an asynchronous parallel system.

4. Parallel synchronous interfaces

The synchronous version of the system described in 3 above, its main application would be high speed data transfer between elements in a tightly coupled multi-processing system employing shared memory. Such systems are becoming increasingly common in surveillance appplication. It is probably advantageous to consider the generation of a standard for such a system before the pressure of project time scales forces various NATO countries to develop their own standards. Back plane or board level interface standards can be derived by taking advantage of de facto standards which exist, such as the Multi Bus (IEEE 796), S-100 Bus (IEEE 696), IEEE 896 Bus, or the IEEE 488 Bus.

5. Variable data interfaces

Another area where interface standardization can be achieved is standard data word formats for subsystems integrated on a multiplex 1553B bus. An example would be navigation position data from an inertial navigation subsystems transmitted on a multiplex data bus. The data word formats would be fixed as to bit values and definition. An example is the draft standard produced under a USAF contract (F33615-80-Ø124).

6. Discrete interfaces

In addition to variable data, a large number of discrete signals need to be transferred around the aircraft. It is therefore necessary to consider an international standard for discrete signals. In the UK this has been considered and Def Stan 00/18(Pt 4) defines such a standard. This standard has already been presented to NATO as Study 3909 AVS. The UK has been invited to produce a draft STANAG.

7. Video data

As more and more electronic display systems and electro optic sensors are employed, it is necessary to consider publishing a standard for the transmission of video data. The advent of colour displays will complicate the issue since within countries comprising NATO there exists three incompatible commercial colour standards. It is therefore essential that a unified NATO video standard is established. This is currently being considered in NATO as Study 3936 AVS.

8. Fibre optics interfaces

The advent of fibre optic communication systems permit much higher bandwidths to be achieved and also greatly improved EMI resistance. Therefore fibre optic data transmission will find increasing use in future systems. It is important to consider NATO fibre optic standards to ensure that compatibility will be achieved. A 1 MHz fibre optic standard is being considered in NATO as Study 3910 AVS, which has produced a draft STANAG. This is the first of a family of future fibre optic standards. Effort should next be directed to defining a wide bandwidth bus standard.

4.3.2.2 Processor systems

The development of processors in recent years has been largely devoted to the realization of a variety of architectures in single LSI chips. The most popular architectures are:-

(a) The classical Von Neuman accumulator based architecture
(b) The general purpose register architecture
(c) The stack oriented architecture
(d) The memory-memory organised architecture.

Architecture types (b), (c) and (d) were developed to make the operation of machines more efficient in comparison to the original Von Neuman approach by reducing the number of memory accesses required and by reducing the number of instructions, by reducing the number of get and put instructions. Since both memory and CPU costs have been reduced dramatically, many of the advantages of these architectures have disappeared. Also, the growing use of high level language and higher processing speeds from VLSI have made the fine details of these architectures of little importance to the vast majority of users.

Developments in software engineering are now focussing attention upon the need for a complete development environment. For example, the Ada Programming Support Environment (APSE) system is being developed as part of the ADA project. The high

costs of such software support environments allied to the need to provide software maintenance facilities over the life of a system means that the vast number of available processors can no longer be supported and that a standard processor is needed.

In the US this has led to the adoption of MIL STD 1750A as the standard USAF processor. This specification defines an instruction set architecture (ISA) which can be provided by a variety of manufacturers in a variety of differing technologies. In addition this ISA should consider a standard interface for non multiplex data bus input/outputs, such as a 16 bit parallel I/O interface. This ISA is now being studied by the UK government as a member of the set of recommended architectures specified by MOD computer policy.

In addition to a general purpose (GP) architecture machine, there are needs for special purpose devices such as:

(a) fast arithmetic processors, possibly working in conjunction with the GP machine. These processors would enable much faster operations to take place;

(b) high speed parallel or array processors to handle such areas as signal processing.

NATO Study 3913 AVS is considering the subject of avionic computer standardization.

4.3.2.3 Memory systems

Memory systems still represent a considerable part of the total cost of a computing system. It is suggested that there is some need for standardization in this area. It is considered unwise to specify a specific memory technology, but it would be feasible to establish a memory interface standard which would specify at least:

(i) Memory access techniques
(ii) Addressing techniques
(iii) Bus structure and communication protocols.

Item (iii) should be related to item 4 of 4.3.2.1 above.

4.3.2.4 Electrical supply interfaces

Power conditioning systems have advanced dramatically in recent years. However, it is still common practice to provide each LRU with its own power conditioning unit. It is suggested that advantages could accrue, if preconditioning was provided in the aircraft. These units would provide a partially stabilised supply. The individual LRUs would then use this supply. This would result in the power supplies of the LRUs being smaller, lighter, cheaper and more reliable. It is suggested that both power supply and power preconditioning standards be established.

4.3.3 Software options and opportunities

4.3.3.1 General

A consideration of software options is complex. However, software costs are becoming more dominant during both development and maintenance. Therefore it is necessary to consider standards in such areas a design techniques, language support systems, operating systems, and the software environment, in addition to standard languages.

In order to examine these other options, it is necessary to consider the requirements for a standard high level language. In general it should be remembered that any computer language, be it assembler or a high level language, is essentially a tool available to the designer to implement his design. The success of a language is dependent on the features it provides the designer to aid in the implementation process.

The software under consideration can be broken down into four main classes. These are:

(a) Support systems, in particular avionics test equipment
(b) Crew training simulators
(c) Mission critical systems, for example, fire control
(d) Safety critical systems, for example, flight control.

Of these systems, test equipment uses ATLAS as a standard. Unfortunately there are many dialects of ATLAS, which limits the usefulness of ATLAS. In the US the IEEE has produced an ATLAS standard which is specified as a standard for USAF programs.

Mission critical systems are those whose failure will affect the success of the mission. Safety critical systems are those whose failure may cause loss of the air vehicle. It should be noted that the safety critical systems form in fact a subset of the mission critical systems on an aircraft. In the past the number of safety critical

systems on an aircraft have been comparatively few. In particular, flight control, engine control and stores management have been considered as safety critical items.

The advent of integrated systems and electronic displays have created more subsystem elements using digital software, which can affect safety. As a result the number of safety critical subsystems is growing. For example, an all-electronic cockpit will force close examination of display subsystems software to ensure that safety critical items, such as loss of display functions, which will place the aircraft in serious jeopardy, are considered. Furthermore, the advent of integrated fire/flight controls will cause critical evaluation of elements in the first control subsystems to ensure they will not become safety critical, to the extent that they influence an unsafe flight path of the aircraft.

The careful distinction that currently exists between mission critical and safety critical items must be strictly maintained in future digital systems, particularly the software of these systems. The reason is that there will be significant differences in verification and validation and certification requirements of the software involved.

An examination of the system requirements will indicate that the following items are required from any support software system and, in particular, from a high level language:

1. Safety
2. Reliability
3. Ability to support structured or top down design
4. Ability to provide concurrency and synchronisation features
5. Economy
6. Ease of use
7. Legibility
8. Good support environment
9. Compatibility
10. Good library procedures.

Considering each of these in turn.

1. Safety

The language design structure and implementation should be such that all conditions are reliably predictable and that no hazardous situations can occur in flight critical systems.

2. Reliability

Software reliability, which can affect safety, has become a major issue in systems and software design, primarily because software has been delegated increasing responsibility in integrating mission and flight critical subsystems. It is essential therefore that the HLL support software, including the compiler and support tools, produce operational code with a high degree of reliability. Without this high degree of reliability, all design efforts to provide system safety and reliability would well be in vain.

3. Ability to support structured or top down design

The value of top down design has been proven many times over in software developments. It follows that any HLL considered for standardization must support top down design methodology.

4. Provision of concurrency and synchronisation functions

Because of the time critical nature of most guidance and control systems, concurrency and synchronisation features are essential in high level languages to support systems implementation.

5. Economy

Any language used should produce compact and economic code.

6. Ease of use

Since any software team is made up of personnel with various skill levels, it is essential that any high level language is easy to understand and use. The language should be easily learned, easily understood, and the effects of all of the constructs available should be clear.

7. Legibility

Software quality requires wide use to be made of structured walk-throughs. It is essential therefore that code produced in a high level language should be extremely legible so that it is easily understood by personnel other than the original writer.

8. Provision of a good support environment

A high level language is only one tool used by the designer. It is essential, therefore, that a good support environment is provided for the language.

9. Compatibility with design procedures

The implementation phase of software design should follow naturally from the design established. As such it is desirable that visibility is maintained and it should be possible to move from the initial design through to the final implementation without losing visibility.

10. Good library procedures

One of the few strengths of FORTRAN is the vast range of mature library routines and defined I/O available within the language. Since real time programming should benefit from such reusable modules, it is essential that these features can be implemented in the language.

No existing language would appear to meet all of these requirements. The earlier languages such as FORTRAN obviously do not support modern structured design techniques and are doubtful in terms of economy. Since ADA is in its infancy, characteristics in terms of reliability, safety, ease of use and good library procedures must be developed and proven.

In practice NATO languages which are closest to providing all of these features are JOVIAL, CORAL and PEARL, largely because they have been in use for sufficient time for their reliability to be established.

A high level language specification is only one of the features need to provide good software, and only one of the options which exist for standardization. In order to establish the other options, it is necessary to consider a total software design, maintenance requirements and the tools needed in each of these areas. These are addressed in the following sections.

4.3.3.2 Options for software standardization

The vast majority of effort in software standardization has been devoted to the standardization of higher order languages. Undoubtedly a higher order language is one of the software elements which needs to be standardized. Given the vast proliferation of languages, it is obviously important to standardize on a language. To reverse this profileration, ADA has been established as a common DOD language.

A higher order language specification is only one of the elements needed in software development. Software tools are also needed throughout the total life cycle of software. The software life cycle is broken down into a number of areas. These are:

(a) Software requirement and specification
(b) Software system design
(c) Software coding and implementation
(d) System testing and integration
(e) Maintenance and support.

Options for standardization exist in most of these areas and, unlike higher order languages, the vast majority are still in the formative stage. It is apparent therefore that options and opportunities exist for a concerted NATO effort to introduce standards across all elements of software development and maintenance. Examining each of these areas in detail:

(a) System description techniques

One of the major sources of error in software is the initial requirement statement for the software. Analysis of such systems as Safeguard in the US has indicated that the vast majority of faults discovered during system integration was due to problems in the initial requirements specification. Part of this problem is due to the fact that such specifications are written in a natural language and, as such, as prone to misinterpretation due to the ambiguity of such languages. English, for example, is prone to ambiguity and therefore specifications written in English are subject to misinterpretation

(b) System design techniques

During the system design phase the software designer is concerned with the structure and architecture of his program. In particular he is concerned with the modules of the program and the interrrelationship between those modules. This design process is greatly aided by having a system design methodology readily available. The functions with which he is concerned are:

(1) The activities of processors within his system;
(2) Communications between these processors;
(3) The degree of parallelism or synchronisation required between the processors or between cooperating processes in any one processor.

Most guidance and control systems consist of parallel processors and of parallel processes inside the processors. A number of system design methodologies exist, for example, SADT, structured hierarchy charts and, in the UK, MASCOT. Such system design tools should form a part of the total support environment needed to develop software. It is suggested that once again an opportunity exists for standardizing upon system description and design languages within NATO.

(c) Coding and implementation

It is during the coding, implementation and maintenance phases that higher order languages are necessary. As was mentioned above, there are a number of requirements which should be met by any higher order language, if it is to be used in guidance and control systems. As also was mentioned above, there would appear to be no existing language which satisfies all of those requirements. ADA can possibly meet most of the requirements, but, because of the early nature of its development, many of the most important features have not been proven. ADA, therefore, would have to be monitored carefully before it is firmly established as a standard guidance and control language - particularly for those areas of a guidance and control system which are safety critical.

A brief examination of ADA indicates that the language is far more extensive than required to code the vast majority of guidance and control subsystems. One possible approach to consider is to establish a proven subset of ADA for those constructs required for safe effective guidance and control. This subset would not be implemented implicitly as a separate ADA language specification compiler. Rather it would use the full compilers already available. The compiler would be given a discrete to identify a high integrity, flight critical software requirement.. The compiler in turn would only execute those portions of the language which have been proven safe for guidance and control purposes. This approach would not infringe the concept that ADA should not be subsetted, which presumably prevents the profileration of a wide variety of ADA subsets and hence loses commonality. This implicit guidance and control usage of ADA would produce code compiled from a fully standard ADA compiler, only some features such as multi tasking, which are flight critical or unsafe, would not be used.

(d) System testing techniques

The system testing phase is largely concerned with detailed testing and integration. The tools used are largely those supplied as part of the system environment. As was mentioned above, one of the requirements of a higher order language is that a good support environment is provided. In this context the support structure proposed for ADA looks extremely powerful and should provide considerable aid in the validation and verification phase of software development. The implicit subset approach suggested above to cope with ADA as a high order language obviously does not invalidate the use of a support environment such as that provided as part of the total ADA package. Certainly the standardization of such a support package would be extremely valuable and once again represents a major opportunity for NATO to develop guidance and control standards.

(e) Maintenance

The technique suggested above would have direct read across into the area of maintenance and support of software during the remainder of its life cycle.

(f) Standards and procedures to provide design disciplines

In addition to the above tools, it is desirable to establish a series of codes of practice and standards to cover such areas as coding, documentation, etc.

4.4 RECOMMENDATIONS

The NATO Standardization Committees (MAS-AVSWP) should consider development of the following standards:

(a) NATO STANAG 3838 should be expanded to include a Data Word Formatting, as recommended in para 4.3.2.5.

(b) NATO STANAG 3838 should be expanded to include discrete signals, such as those included in UK Def Stan 0018(Part IV).

(c) High Speed MUX bus standards should also be developed to accommodate the requirements outlined in para 4.3.2.6 (video) and 4.3.2.7 (fibre optics).

(d) The input/output definitions in MIL STD 1750A should be expanded to include a 16 bit parallel I/O interface as per para 4.3.2.2.

(e) Back plane/board interface standards described in para 4.3.2.1.4 should be developed, with due caution, to ensure these hardware standards will be technology transparent and technology independent.

(f) Power supply and power preconditioning standards should be considered, as recommended in para 4.3.2.4.

(g) A NATO STANAG covering the ATLAS test language should be developed to provide a standard NATO equipment language language.

(h) The Terminology and Nomenclature in Chapter III should be developed into a NATO STANAG.

4.5 In addition to the recommendation in para 4.4, an AGARD working group should be established to examine system description techniques, system design techniques, high level language requirements and support tools for avionic systems. This work should in particular address the development of the new ADA language.

REPORT DOCUMENTATION PAGE			
1. Recipient's Reference	2. Originator's Reference AGARD-AR-178	3. Further Reference ISBN 92-835-1428-9	4. Security Classification of Document UNCLASSIFIED
5. Originator	Advisory Group for Aerospace Research and Development North Atlantic Treaty Organization 7 rue Ancelle, 92200 Neuilly sur Seine, France		
6. Title	DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS		
7. Presented at			
8. Author(s)/Editor(s)	Various Edited by Louis J.Urban*		9. Date July 1982
10. Author's/Editor's Address	*Technical Director, Deputy for Avionics Control Aeronautical Systems Division Wright-Patterson AFB Ohio 45433, USA		11. Pages 114
12. Distribution Statement	This document is distributed in accordance with AGARD policies and regulations, which are outlined on the Outside Back Covers of all AGARD publications.		
13. Keywords/Descriptors	Microcircuits Microprocessors High level languages (HLL) Microprocessor architecture, VHSI, VLSI		

14. Abstract

The objectives of Working Group 06 were to obtain a better understanding of micro-processor technology developments, and of architecture configurations for NATO guidance and control systems. A lexicon of terms and nomenclatures is provided as well as considerations on standardization options and opportunities.

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under the Chairmanship of Louis J.Urban.

AGARD Advisory Report No.178 Advisory Group for Aerospace Research and Development, NATO DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS Edited by Louis J.Urban Published July 1982 114 pages The objectives of Working Group 06 were to obtain a better understanding of microprocessor technology developments, and of architecture configurations for NATO guidance and control systems. A lexicon of terms and nomenclatures is provided as well as considerations on standardization options and opportunities. P.T.O.	AGARD-AR-178 Microcircuits Microprocessors High level languages (HLL) Microprocessor architecture, VHSI, VLSI	AGARD Advisory Report No.178 Advisory Group for Aerospace Research and Development, NATO DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS Edited by Louis J.Urban Published July 1982 114 pages The objectives of Working Group 06 were to obtain a better understanding of microprocessor technology developments, and of architecture configurations for NATO guidance and control systems. A lexicon of terms and nomenclatures is provided as well as considerations on standardization options and opportunities. P.T.O.	AGARD-AR-178 Microcircuits Microprocessors High level languages (HLL) Microprocessor architecture, VHSI, VLSI
AGARD Advisory Report No.178 Advisory Group for Aerospace Research and Development, NATO DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS Edited by Louis J.Urban Published July 1982 114 pages The objectives of Working Group 06 were to obtain a better understanding of microprocessor technology developments, and of architecture configurations for NATO guidance and control systems. A lexicon of terms and nomenclatures is provided as well as considerations on standardization options and opportunities. P.T.O.	AGARD-AR-178 Microcircuits Microprocessors High level languages (HLL) Microprocessor architecture, VHSI, VLSI	AGARD Advisory Report No.178 Advisory Group for Aerospace Research and Development, NATO DISTRIBUTED MICRO-PROCESSOR APPLICATIONS TO GUIDANCE AND CONTROL SYSTEMS Edited by Louis J.Urban Published July 1982 114 pages The objectives of Working Group 06 were to obtain a better understanding of microprocessor technology developments, and of architecture configurations for NATO guidance and control systems. A lexicon of terms and nomenclatures is provided as well as considerations on standardization options and opportunities. P.T.O.	AGARD-AR-178 Microcircuits Microprocessors High level languages (HLL) Microprocessor architecture, VHSI, VLSI

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under the Chairmanship of Louis J.Urban.

ISBN 92-835-1428-9

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under Louis J.Urban.

ISBN 92-835-1428-9

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under the Chairmanship of Louis J.Urban.

ISBN 92-835-1428-9

This publication, requested by the Guidance and Control Panel of AGARD, comprises four chapters, and is the result of work accomplished by GCP/WG.06 members under Louis J.Urban.

ISBN 92-835-1428-9

B247
9

AGARD

NATO OTAN
7 RUE ANCELLE - 92200 NEUILLY-SUR-SEINE
FRANCE
Telephone 745.08.10 - Telex 610176

DISTRIBUTION OF UNCLASSIFIED AGARD PUBLICATIONS

AGARD does NOT hold stocks of AGARD publications at the above address for general distribution. Initial distribution of AGARD publications is made to AGARD national delivery agencies (normally one copy per country). Individual copies are sometimes available from these Centres, but if not may be purchased in Microfiche or Photocopy form from the sources listed below.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Coordonnateur AGARD – VSL
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evere, 1140 Bruxelles

CANADA
Defence Science Information Services
Department of National Defence
Ottawa, Ontario K1A 0K2

DENMARK
Danish Defence Research Board
Gamlehavnsvej 56
Copenhagen Ø

FRANCE
O.N.E.R.A. (Direction)
29 Avenue de la Division Leclerc
92320 Châtillon sous Bagneux

GERMANY
Fachinformationszentrum Energie,
Physik, Mathematik GmbH
Kernforschungszentrum
D-7514 Eggenstein-Leopoldshafen 2

GREECE
Hellenic Air Force General Staff
Research and Development Directorate
Holargos, Athens

ICELAND
Director of Aviation
c/o Flugrad
Reykjavik

ITALY
Aeronautica Militare
Ufficio del Delegato Nazionale all'AGARD
3, Piazzale Adenauer
Roma/EUR

LUXEMBOURG
See Belgium

NETHERLANDS
Netherlands Delegation to AGARD
National Aerospace Laboratory, NLR
P.O. Box 126
2600 A.C. Delft

NORWAY
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
N-2007 Kjeller

PORTUGAL
Direcção de Serviços de Material
Da Força Aérea
Rua da Escola Politécnica 42
Lisboa
Attn: AGARD National Delegate

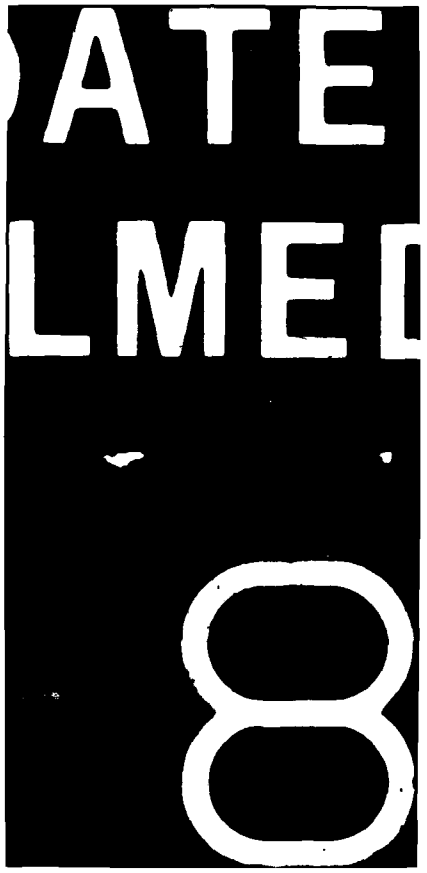
TURKEY
Department of Research and Development (ARGE)
Ministry of National Defence, Ankara

UNITED KINGDOM
Defence Research Information Centre
Station Square House
St. Mary Cray
Orpington, Kent BR5 3RE

UNITED STATES
National Aeronautics and Space Administration (NASA)
Langley Field, Attn: 180
Hampton, Virginia 23665

THE UNITED STATES NATIONAL DISTRIBUTION CENTRE (NASA) DOES NOT HOLD
STOCKS OF AGARD PUBLICATIONS, AND APPLICATIONS FOR COPIES SHOULD BE MADE DIRECT TO THE NASA
DEPOSITORY AT THE ADDRESS GIVEN ABOVE. PUBLICATIONS ARE NOT NORMALLY AVAILABLE FROM PUBLISHING SALES AGENTS.

[redacted]


ATE
LME
8